REPUBLIQUE DU CAMEROUN
Paix-Travail-Patrie
****
UNIVERSITE DE YAOUNDE 1
****
CENTRE DE RECHERCHE ET DE FORMATION DOCTORALE EN SCIENCES, TECHNOLOGIE ET GEOSCIENCES
****
UNITE DE RECHERCHE ET DE FORMATION DOCTORALE EN PHYSIQUE ET APPLICATIONS
****
B.P : 812 Yaoundé
Email : crfd@uy1.uninet.cm

REPUBLIC OF CAMEROON
Peace-Work-Fatherland
****
THE UNIVERSITY OF YAOUNDE 1
****
POSTGRADUATED SCHOOL OF SCIENCE, TECHNOLOGY AND GEOSCIENCES
****
RESEARCH AND POSTGRADUATE TRAINING UNIT FOR PHYSICS AND APPLICATIONS
****
P.O BOX : 812 Yaoundé

Web Site : www.uy1researchstg.cm

Laboratoire d'Énergie, des Systèmes Électriques et Électroniques
Laboratory of Energy, Electrical and Electronic Systems

CONTRIBUTION A L'ÉTUDE EXPÉRIMENTALE D'UN SYSTÈME DE CRYPTOGRAPHIE DE SIGNAUX : APPLICATION AUX IMAGES MÉDICALES

Thèse présentée en vue de l'obtention du Diplôme de Doctorat/Ph.D de Physique

Option : **Systèmes Électriques et Électroniques**

par

HEUCHEUN YEPDIA Lee Mariel

Matricule : **13R2715**

Master of Siences en Physique

Devant le jury composé de :

Président :	**NDJAKA Jean-Marie Bienvenu**	**Professeur**	
Rapporteur :	**TIEDEU Alain**	**Professeur**	**UYI**
Membres :	**WOAFO Paul**	**Professeur**	**UYI**
	ESSIMBI ZOBO Bernard	**Professeur**	**UYI**
	MBOUPDA PONE Justin Roger	**Maître de Conférences**	**UDs**
	BODO Bertrand	**Maître de Conférences**	**UYI**

Année 2022

UNIVERSITE DE YAOUNDE I

UNIVERSITY OF YAOUNDE I

FACULTE DES SCIENCES

FACULTY OF SCIENCES

DEARTEMENT DE PHYSIQUE

DEPARTMENT OF PHYSICS

ATTESTATION DE CORRECTION DE LA THÈSE DE DOCTORAT/Ph.D

Nous, Professeur **NDJAKA Jean-Marie Bienvenu** et Professeur **MBOUPDA PONE Justin Roger**, respectivement Président du jury et Examinateur de la Thèse de Doctorat/Ph.D de Monsieur **HEUCHEUN YEPDIA Lee Mariel**, Matricule **13R2715**, préparée sous la supervision du Professeur **TIEDEU Alain**, intitulée : « **Contribution à l'étude expérimentale d'un système de cryptographie de signaux : application aux images médicales** », soutenue le **Mardi 04 Janvier 2022**, en vue de l'obtention du grade de Docteur/Ph.D en Physique, Spécialité **Systèmes Électriques et Électroniques**, attestons que toutes les corrections demandées par le jury de soutenance ont été effectuées.

En foi de quoi, la présente attestation lui est délivrée pour servir et valoir ce que de droit.

Fait à Yaoundé, le 05 JAN 2022

Examinateur

Pr. MBOUPDA PONE Justin Roger

Le Président du Jury

Pr. NDJAKA Jean-Marie Bienvenu

Le Chef de Département de Physique

UNIVERSITE DE YAOUNDE I FACULTE DES SCIENCES DEPARTEMENT DE PHYSIQUE Le Chef de Département

Pr. NDJAKA Jean-Marie B.

Professeur

CONTRIBUTION À L'ÉTUDE EXPÉRIMENTALE D'UN SYSTÈME DE CRYPTOGRAPHIE DE SIGNAUX : APPLICATION AUX IMAGES MÉDICALES

présenté et soutenue par:

HEUCHEUN YEPDIA Lee Mariel

Master of sciences en Physique

Sous la Direction de

TIEDEU Alain

Professeur

UNIVERSITE DE YAOUNDE I

Janvier 2022

Dédicace

A mes parents Yepdia Daniel et Tchapnda Rose.
A ma merveilleuse épouse Sonkoue Eliane.
A mes enfants.
Aux grandes familles Yepdia et Ndjiofack.

Remerciements

Je suis reconnaissant au Dieu tout puissant qui m'a accordé sa grâce pour réaliser ces travaux.

Je remercie particulièrement le Professeur Alain Tiedeu pour avoir accepté encadrer mes travaux, pour ses conseils et son soutien au sens paternel.

Je remercie également Madame Tiedeu Barbara pour ses encouragements et conseils.

Je remercie le chef de Département de Physique le Professeur Ndjaka Jean Marie pour sa bienveillance et ses services rendus.

Je remercie le Professeur Kofane Timoléon pour ses enseignements et conseils.

Je formule mes remerciements à l'endroit du Directeur de laboratoire le Professeur Essimbi Zobo Bernard pour son encadrement.

Je remercie les Enseignants du Département de Physique de la Faculté de Sciences de l'Université de Yaoundé I pour leur formation académique et leur encadrement.

Je tiens à remercier tous les enseignants du département, notamment les Professeurs Eyebé Fouda Jean Sire, Bodo Bertrand, Biya Motto pour leurs enseignements, encadrement et encouragements.

Je remercie de tout cœur les aînés du Laboratoire Professeur Mboupda Poné Justin R, Docteur Kenfack Gutenbert, Docteur Yannick Abanda, Professeur Kom Guillaume, Docteur Kamdeu Pascal pour leur aide et conseils.

Je remercie mes camarades de promotion et ceux du laboratoire, Tamba Patrick, Tchuente Stéphane, Von Ricardo, Djembi Gaëlle, Nguenou Joël, Tampa Romaric, Makem Mimosette, Fokou Adolphe, Djomo Alain, pour leur aide, bon sens d'amitié et de collaboration.

Je remercie l'Ecole Nationale Supérieure Polytechnique, l'ENSET d'Ebolowa pour leur soutien pendant mon cycle de recherche.

Je remercie le programme ERMIT (Entrepreneurship, Resources, Management, Innovation and Technologies) pour la bourse qu'il m'a offerte pour effectuer mes recherches en mobilité à l'Ecole Nationale des Ingénieurs de Tunis (ENIT), en Tunisie au cours de l'année académique 2019-2020).

Par ailleurs, je remercie l'ENIT pour tout le soutien et l'hospitalité qui nous a été accordé pendant notre séjour de recherche en Tunisie.

Je remercie particulièrement le Professeur Zied Lachiri de l'ENIT pour l'attention qu'il m'a

accordée ainsi qu'une bonne collaboration.

Je remercie de tout cœur Professeur Boyomo Marthe pour ses conseils et son soutien.

Je remercie l'école Doctorale et tous les enseignants ayant contribué à l'amélioration de la qualité de ce document.

Je suis reconnaissant à l'endroit de tous les membres du Jury de soutenance de cette thèse pour leurs contributions enrichissantes et remarques constructives.

Je remercie mes parents Yepdia Daniel et Tchapnda Rose, mes beaux-parents Feu Ndiofack Francois et Kenne Bertine pour l'éduction qu'ils m'ont communiquée, leurs encouragements et conseils.

Je remercie grandement mon épouse Sonkoue Eliane pour tout son soutien pendant ces mes moments de travail, et pour sa bonne gestion de la maison en mon absence.

Je remercie mes frères et sœurs, Docteur Yepdia Leundjeu Walter, Tchuisseu Idylle Claver, Mbadjoun Mireine, Djeumegni Julien Sorel, Ndjiofack Flavine, Tchantchou Dandy, Révérend Tchuigoua Willy, Manfouo Ndjiofack Oranelle, Stanislas pour leur aide, conseils et réconfort.

Je remercie particulièrement Mbadjoun Mireine pour le temps qu'elle a consacré à la relecture de ce document.

J'adresse ma reconnaissance aux groupes du Culte d'enfants de l'Eglise Evangélique du Cameroun de Melen et au groupe Family-Life Cameroun pour tout leur accompagnement et soutien.

Je remercie le couple Nyimga pour leur encadrement, aide et conseils.

Sans être exhaustif, je remercie tous ceux qui de près ou de loin ont contribué à la réalisation de ce document.

Table des matières

Table des figures

Liste des tableaux

Sigles et acronymes

ADN :	Acide Désoxyribonucléique
AES :	Advanced Encryption Standard
ANN :	Artificial Neural Network
CBC :	Cipher Block Chaining
CEM :	Carte Chaotique Economique
CFB :	Cipher FeedBack
CIE :	Composite Image Element
CTR :	Counter
DCT :	Discrete Cosine Transform
DES :	Data Encryption Standard
DRPE	Double Random Phase Encoding
DWT :	Discrete Wavelet Transform
ECB :	Electronic Code Book
ENIT :	Ecole Nationale des Ingénieurs de Tunis
ENSET :	Ecole Nationale Supérieure d'Enseignement Technique
ERMIT :	Entrepreneurship, Resources, Management, Innovation and Technologies
FPGA :	Field Programmable Gate Array
GNSA :	Générateur de Nombres Pseudo-Aléatoires
ICA :	Independant Component Analysis
IDEA :	International Data Encryption Algorithm
IHS :	Intensity Hue Saturation
J-C :	Jésus-Christ
MATLAB :	Matrix Laboratory
MSE :	Mean Square Error
NIST :	National Institute of Standards and Technology
NMSE :	Normalised Mean Square Error
NPCR :	Number of Pixel Change Rate
OFB :	Output Feedback
PCA :	Principal Component Analysis

PCNN : Pulse Coupled Neural Network
PSNR : Peak Signal Noise to Ratio
PWLCM : Piece Wise Linear Chaotic Map
RMS : Root Mean Square
RSA : Rivest, Shamir et Adleman
SBWP : Product Space-Bandwith
SFCE : Simultaneous Fuison, Compression and Encryption
SHA : Secure Hash Algorithm
UACI : Unified Average Changing Intensity

Résumé

Ces dernières décennies, la télémédecine a connu un essor considérable grâce au développement des technologies de l'information et de la communication. De nombreuses informations sont ainsi échangées entre les praticiens de la santé. Une grande partie de ces informations est constituée d'images médicales dont plusieurs contiennent des informations sensibles des patients et qui nécessitent de la confidentialité. Dans le souci d'assurer la sécurité d'images lors de leur transmission d'un point à l'autre, plusieurs algorithmes de cryptage sont développés, parmi lesquels ceux basés sur la fusion ou mixage d'images.Ces dernières sont de plus en plus mis à contribution. En fait, la majorité des systèmes de fusion d'images vise à fournir une information plus riche en vue d'une meilleure prise de décision. Cependant, ces dernières années, des techniques de fusion d'images ont plutôt été développées pour le cryptage d'images. Nous proposons dans ce document deux algorithmes de chiffrement à clés secrètes basés sur la fusion d'images. Par la suite, un début d'implémentation du premier algorithme est mené sur la carte intégrée à microprocesseur STM32F407ZET6 afin d'apprécier une possible mise en oeuvre du cryptosystème sur cette carte. Ces deux crypto-systèmes proposés sont bâtis suivant la structure confusion-diffusion. Le premier utilise les cartes chaotiques Henon, logistique-may et logistique-sine dans le processus de chiffrement, où deux images sont brouillées séparément, puis mixées afin de produire deux images hybrides. Le second réalise la fusion d'images cibles à deux niveaux : la transformation discrète en cosinus (DCT en anglais) est utilisée en premier temps pour fusionner les images cibles en deux images multiplexées, dont chacune sera permutée au niveau des blocks. Le second niveau de fusion est effectué par une relation non linéaire inspirée du modèle de Kramer. Les cartes may-gaussienne et gaussienne-gompertz sont utilisées comme générateurs de nombres pseudo-aléatoires dans ce second algorithme. Les tests d'évaluation des performances des algorithmes proposés ont été effectué sur des images standards de la communauté scientifique, et ensuite sur des images médicales. Les résultats obtenus après simulations, analyses et comparaison avec des algorithmes de la littérature montrent qu'ils sont satisfaisants en termes de robustesse, rapidité de temps d'exécution, quantité d'images à crypter et qualité d'images reconstruites. De plus, les algorithmes proposés présentent une structure simple et sont adaptés à une implémentation dans une architecture matérielle telle que la carte STM32F407ZET6.

Mots clés : Système de cryptographie, fusion d'images ; cryptage, image médicale

Abstract

Over the last few decades, telemedicine has grown considerably thanks to the development of information and communication technologies. A lot of information is exchanged between health care practitioners. Much of this information consists of medical images, many of which contain sensitive patient information and require confidentiality. In order to ensure the security of images during their transmission from one point to another, several encryption algorithms have been developed, among which those based on the fusion or mixing of images are increasingly used. In fact, the majority of image fusion systems aim at providing richer information for better decision making. However, in recent years, image fusion techniques have been developed more for image encryption. In this paper, we propose two secret key encryption algorithms based on image fusion. Subsequently, a beginning of implementation of the first algorithm is led on the STM32F407ZET6 microprocessor embedded board in order to appreciate a possible implementation of the cryptosystem on this card. The first one uses the Henon, logistic-May and logistic-sine chaotic maps in the encryption process, where two images are scrambled separately and then mixed to produce two hybrid images. The second performs two-level target image fusion : the discrete cosine transform (DCT in english) is used first to merge the target images into two multiplexed images, each of which is permuted at the block level. The second level of fusion is performed by a non-linear relation inspired by the Cramer model. may-gaussian and gaussian-gompertz maps are used as pseudo-random number generators in this second algorithm. The performance evaluation tests of the proposed algorithms have been performed on standard images of the scientific community, and then on medical images. The results obtained after simulations, analyses and comparison with algorithms of the literature, show that they are satisfactory in terms of robustness, speed of execution time, quantity of images to encrypt and quality of reconstructed images. Moreover, the proposed algorithms present a simple structure, and are adapted to an implementation in a hardware architecture such as the STM32F407ZET6.

Key words : Cryptography system, image fusion, encryption, medical image

Introduction générale

En réponse au besoin d'assurer les échanges en termes de communications entre différentes personnes et équipements, les nouvelles technologies et l'Internet sont à ce jour des moteurs favorisant l'échange et le stockage d'un grand nombre de données très sensibles et d'origine diverses, à travers des canaux de communication. Le besoin d'intégrité, d'authenticité, non-repudiation et surtout de confidentialité lors de l'échange de ces données a favorisé le développement de la cryptographie. Cette science produit les outils et moyens pour sauvegarder les données contre les voies non autorisées et assurer le transfert sécurisé d'‘informations d'un point à l'autre. Elle est d'autant plus importante de nos jours, car il est nécessaire de protéger les transactions bancaires des individus et entreprises, les dossiers médicaux des patients, les informations militaires des nations, les communications privées des hautes personnalités, etc. Parmi les informations produites et échangées dans les réseaux de communication et applications multimédia, les images numériques, plus précisément les images médicales occupent une place de choix, car leur transmission et leur stockage fait l'objet d'un grand nombre d'opérations en ce 21e siècle. Par exemple, dans les zones rurales, plusieurs dossiers sensibles des patients sont transmis vers d'autres pays pour diagnostic et analyse, à travers des voies de communications peu sécurisées. Également, dans les centres hospitaliers équipés, les opérations à distance des patients se font de plus en plus par les spécialistes de la santé. Par conséquent, la nécessité de sécuriser ces images devient capitale et essentielle.

Présentation et situation du thème de recherche

Le développement de la cryptanalyse et l'amélioration de la puissance des processeurs des calculateurs numériques contribuent de plus en plus au développement de nouveaux algorithmes de cryptage d'images par les chercheurs. Plusieurs crypto-systèmes ont été développés par le passé pour le chiffrement de données, et bon nombre d'entre eux étaient basés sur les techniques classiques telles que DES, AES, IDEA, RSA, etc. [1]. Ces techniques de chiffrement ont connu un grand essor pendant et après la deuxième guerre mondiale pour des fins de sécurité militaire. Dans ce contexte, ces algorithmes étaient dédiés à la sécurité des informations textuelles ; c'est ce qui peut justifier leurs limites à chiffrer des images ou des vidéos. En effet, les images sont caractérisées par une forte corrélation entre les pixels adjacents, une grande quantité de

données, une forte redondance des valeurs et une faible entropie [2], [3]. De plus, la taille d'un fichier image est plus grande que les autres données numériques comme le texte et l'audio, et le cryptage d'images numériques nécessite de grandes quantités de calcul, d'où la nécessité de concevoir des algorithmes adaptés pour gérer ce type de données. Après la deuxième guerre mondiale, la cryptographie s'est modernisée, et plusieurs auteurs ont proposé des techniques de chiffrement d'images utilisant les cartes chaotiques [4]–[13], les transformations [14]–[18], les algorithmes évolutifs [19], la séquence de l'acide désoxyribonucléique ,ADN [14], [20]–[24] et bien d'autres [16], [25]–[30]. Ces dernières décennies, l'étude des systèmes chaotiques par les chercheurs a révélé qu'ils sont adaptés pour la construction des crypto-systèmes robustes. En effet, leurs propriétés telles que l'ergodicité, la sensibilité aux conditions initiales et aux paramètres de contrôle, le comportement aléatoire et le déterminisme permettent de réaliser les opérations de confusion et diffusion recherchées dans les applications en cryptographie. Les atouts de tels systèmes justifient l'existence d'un grand nombre d'algorithmes de chiffrement basés sur les systèmes à base de cartes chaotiques [20]. Toutefois, il est bon de relever que plusieurs crypto-systèmes développés utilisant les cartes chaotiques ont été cryptanalysés, soit à cause de leur faible espace de clé, ou alors l'insensibilité à l'image en clair ou à la clé, ou enfin le comportement chaotique limité, etc. [31]–[42] pour ne citer que ceux-là.

En parcourant la littérature, il y'a quelques années, l'on a constaté qu'il était possible de crypter les images par fusion ou mixage. En fait, le cryptage par fusion d'images présente de bons avantages, notamment la possibilité de combiner plusieurs images en une avant analyse ou transmission, et de recombiner les images sources à partir de l'image cryptée sans pertes d'informations. Auparavant, les outils de fusion étaient utilisés en traitement de signal comme outil d'aide à la décision dans diverses applications telles que la reconnaissance de formes, la classification, le contrôle automatisé, etc. [43]–[47].
En conséquence, plusieurs auteurs ont proposé ces dernières années des algorithmes de chiffrement d'images intégrant le concept de fusion, soit en mode post-traitement, soit comme outil principal de cryptage. En guise d'exemple, Liu X. et al.[48] ont proposé un algorithme de cryptage simultané d'images basé sur la fusion et détection-compression. Les mêmes auteurs ont proposé une année plus tôt [49] un crypto-système fusionnant des images en exploitant le chaos, la détection-compression et la transformée de Fourier Fractionnaire. Dans ces deux algorithmes précédents, la fusion est utilisée comme étape de prétraitement avant la phase principale de chiffrement. Dans la même dynamique, Yi Q. et al. [50] ont proposé un algorithme de chiffrement optique d'images, basé sur la fusion spectrale et des opérations non linéaires. Dans le processus de chiffrement, la DCT est appliquée aux différentes images cibles, et les spectres sont combinés en une image qui sera multipliée par un masque suite à un ensemble d'opérations non-linéaires. Malgré la simplicité de la structure de l'algorithme, il présente des

faiblesses au niveau de la qualité d'images reconstruites ainsi que le problème de sensibilité à la clé. Isha et al. [51] ont développé un algorithme de chiffrement d'images dans lequel les images sont fusionnées par la Transformée en Ondelettes, et l'image obtenue est associée à des masques de phases. Le crypto-système est asymétrique, car il offre de bonnes performances en termes de robustesse et espace de clé, mais présente des failles quant au temps d'exécution qui est long, ainsi que la qualité des images décryptées qui est tributaire du nombre d'images à chiffrer.

Récemment, Shi D. et al. [52] ont proposé un nouvel algorithme de cryptage et compression simultané d'images utilisant la méthode optique du détecteur de pixel. Selon cette approche, les images sources sont multiplexées au niveau du pixel par un dispositif optique utilisant le principe de réflexion. L'algorithme est adapté au cryptage d'images en temps réel, mais a un espace de clé réduit, par conséquent vulnérable aux attaques exhaustives. Également, il n'offre pas un bon compromis entre rapport de compression et qualité d'images reconstruites. Ayman A. et Mansour A. [53] ont développé une nouvelle approche de cryptage basée sur l'utilisation des méthodes de cryptage par filtrage fréquentiel et analyse des composantes indépendantes « ICA » (Independent Component Analyses). Les auteurs utilisent les images très proches d'une séquence vidéo qui sont cryptées individuellement par un masque de phase, puis sont combinées de manière linéaire par une relation effectuant la combinaison linéaire de toutes les images. Les trois images chiffrées obtenues sont ensuite envoyées séparément par différents canaux. L'algorithme proposé offre de bonnes performances en termes de robustesse et peut chiffrer de grandes quantités d'images. Toutefois, après décryptage, la qualité d'images reconstruites n'est pas satisfaisante

Des algorithmes de fusion ou mixage d'images ont été proposés dans le domaine spatial pour corriger les limitations au niveau de la qualité d'images décryptées et la robustesse, observées dans les algorithmes précédents où la fusion est effectuée dans le domaine spectral. Dans cet ordre, Maher J. et Ayman A. [54] ont proposé un nouvel algorithme pour renforcer les failles de l'algorithme de fusion, compression et cryptage simultané d'images [55] en termes de temps d'exécution, bande passante et robustesse. Dans ce nouvel algorithme, les auteurs utilisent une approximation de la Transformée Discrète en Cosinus pour réduire le temps d'exécution à la phase de fusion d'images. Par la suite, les cartes chaotiques de Henon et de Skew-Tent sont utilisées respectivement pour réaliser la confusion au niveau des lignes, puis des colonnes et assurer la phase de diffusion. L'algorithme proposé est robuste, et permet de crypter plusieurs images. Deux autres algorithmes de chiffrement simultané de plusieurs images ont été proposé par Xiaoqiang Z. et Xuesong W. [56], [57] : le premier est basé sur le mixage d'éléments d'images et la permutation, et le second effectue le mixage d'images par les cartes chaotiques. Ces deux systèmes introduisent le concept d'éléments d'image pure et éléments d'image mixée, et présentent l'avantage de chiffrer plusieurs types d'images simultanément, avec de bonnes

performances de robustesse et avec un temps d'exécution réduit. Au cours du processus de chiffrement, les images originales sont combinées en une grande image qui est par la suite subdivisée en plusieurs blocs de petite taille. Ces blocks sont mixés, et les pixels de l'image brouillée obtenue sont diffusés afin d'obtenir l'image chiffrée. Malgré les performances de ces algorithmes, il se pose le problème de taille de l'image mixée qui devient très grande lorsque le nombre d'images à chiffrer est considérable. En guise d'optimisation de ces algorithmes, l'image mixée cryptée contenant toutes les images cibles doit être compressée avant transmission, ou un modèle de crypto-compression peut être développé pour chiffrer cet ensemble d'images. Un autre algorithme de mixage d'images a été proposé par Gui-Liang et al. [58], basé sur les courbes elliptiques. Cet algorithme introduit le concept d'images composite. Les simulations expérimentales attestent que l'algorithme possède un large espace de clé, un bon niveau de sécurité dans les réseaux de communication, mais nécessite un temps d'exécution élevé ; de ce fait, n'est pas approprié pour les applications en temps réel. Ayman A. et al. [59] ont proposé un crypto- système qui mixe plusieurs images de différentes tailles et types, en utilisant une procédure de brouillage de pixels combinée aux boites S de substitution. Les simulations et analyses montrent que l'algorithme a des performances acceptables, mais doit être optimisé en termes de robustesse, car vulnérable aux attaques à force brute. Certains auteurs ont aussi utilisé la transformation de « Jigsaw », et d'autres la carte du chat d'Arnold pour effectuer une auto-fusion d'images, mais, après analyse de résultats, des faiblesses au niveau de l'espace de clés, ainsi que la structure de l'algorithme ont été relevés [60], [61].
Après analyse de plusieurs cryptosystèmes présents dans la littérature, plusieurs d'entre eux ne prennent pas compte lors de la conception de leur système, de la sécurité des données lors de leur transmission vers le système récepteur. En fait, dans la plupart des systèmes de cryptage existants, tant qu'une image est interceptée, ou lorsqu'on a accès au système de cryptage, il est possible d'effectuer une cryptanalyse du code en utilisant les principaux tests connus (attaque en texte clair connue, attaque en texte clair choisie, attaque différentielle, etc.). Une des principales raisons qui rend cette cryptanalyse possible est que les données traitées constituent une source cohérente d'informations recherchées ou cachées. Il est possible de réduire l'effet de la cryptanalyse sur les cryptosystèmes, notamment en proposant des cryptosystèmes construits selon une bonne structure, et dont les images cryptées sont sources d'informations non cohérentes. Dans ce cas, les images cibles peuvent être combinées ou fusionnées avant d'être cryptées, puis transmises par plusieurs canaux de communication, ou dans le même canal par trames. Ainsi, pour qu'une image soit décryptée par un hacker, il lui faudra en plus de maîtriser la clé, de disposer de toutes les autres images cryptées.

Fort de ce qui précède, l'on peut observer après une brève revue de littérature sur les techniques de cryptages basés sur la fusion ou le mixage d'images qu'elles présentent d'énormes

atouts, mais aussi plusieurs limites (failles). En termes d'avantages :

- Elles sont récentes et offrent différentes approches d'implémentation ;
- Permettent de chiffrer plusieurs images de différents types et tailles simultanément ;
- Sont d'un apport considérable pour la transmission sécurisée de grandes quantités d'images à travers les canaux de communication dans un environnement non sécurisé ;
- Peuvent permettre un bon compromis entre quantité d'images transmises et robustesse du système ;
- Sont aisément implémentables dans une architecture matérielle ;
- Peuvent pour certains garantir la robustesse et l'efficience ;
- Peuvent assurer la sécurité double des informations à transmette : au niveau du contenu et dans le canal de transmission.

En revanche, les algorithmes de cette typologie présentent plusieurs limitations, telles que évoquées dans les paraphes en amont. Les principales failles relevées sont :

- Les failles de sécurité ([55],[62]) ;
- La qualité des images décryptées ([48],[50],[52],[63]) ;
- Temps d'exécution trop long ([58]) ;
- Mauvais compromis entre quantité d'images cryptées et qualité d'images reconstruites ;
- Mauvais compromis entre rapport de compression et robustesse ;
- Insensibilité au texte en clair et ou à la clé de chiffrement ([51]) ;
- Un faible espace de clé.

Question de la recherche et contribution

Les algorithmes de cryptage d'images basés sur la fusion ou le mixage d'images nous semblent malgré leurs limites une bonne voie à explorer pour les chercheurs, d'où la question : Est-il possible de construire un crypto-système basé sur la fusion ou le mixage d'images qui soit robuste, rapide, simple à implémenter dans une architecture matérielle, et qui offre un bon compromis entre quantité d'images chiffrées et qualité d'images reconstruites ?
En guise de solutions à ce problème, nous proposons dans ce travail deux algorithmes basés sur la fusion/mixage d'images, et proposons un début d'implémentation matérielle du premier algorithme sur carte intégrée STM32F407ZET6 :

- Le premier algorithme consiste à crypter deux (voire plusieurs) images par fusion avec une approche hybride. Chacune des images est chiffrée séparément par un algorithme simple, l'un basé sur l'architecture confusion-diffusion, et l'autre utilisant la technique de brouillage. Dans les deux cas, les cartes chaotiques sont utilisées comme générateurs de nombres pseudo-aléatoires. Les deux images chiffrées sont alors combinées à la phase de diffusion par une relation exploitant la règle de Kramer.

- Le deuxième algorithme effectue la fusion de plusieurs images en exploitant la DCT, et des cartes chaotiques construites à base des cartes 1D (Tent, Logistique, May et Gompertz). Les images cibles sont fusionnées à deux niveaux, par un processus de confusion assuré par la permutation des blocks d'images, et la phase diffusion réalisée par un système d"équations non linéaire.
- L'environnement de developpement de la carte STM32F407ZET6 est illustré ainsi que sa connexion avec ses périphériques.

Dans le premier chapitre, nous ferons une présentation sommaire sur l'historique de la cryptographie, sa terminologie et ses concepts de base. Par la suite, nous distinguerons les familles cryptographiques, les différents modes de chiffrement et les généralités sur la fusion d'images. Le chapitre se poursuit avec un état des contributions sur les crypto-systèmes exploitant la fusion ou le mixage d'images dans leur processus de chiffrement. Il s'achève par la présentation de quelques procédés de cryptanalyse.

Le chapitre deux débute avec la description d'outils physiques et mathématiques que nous avons utilisé dans l'élaboration des algorithmes proposés. Ces outils présentent succinctement des éléments sur les systèmes dynamiques, les tests d'analyse du caractère aléatoire des nombres et les cartes chaotiques utilisées. Les trois algorithmes proposés sont ensuite décrits en détail. Le chapitre se referme par la présentation d'outils d'évaluation des performances d'un bon crypto système.

Le troisième chapitre porte sur les résultats et analyses. Tout d'abord, des éléments présentant le caractère aléatoire des cartes chaotiques utilisées sont illustrés, la présentation de la base de données d'images utilisées est faite. Cette base contient des images de différentes taille, et principalement des images médicales. Ensuite, l'évaluation des algorithmes de chiffrement proposés est faite à l'aide des métriques évoquées au chapitre deux. Enfin, le chapitre se referme par la description du processus d'implémentation matérielle du premier algorithme sur la carte intégrée STM32F407ZET6.

Le présent document s'achève avec une conclusion générale, suivie des perspectives qui présentent les autres voies à explorer concernant le thème abordé.

Chapitre I

État de l'art sur le cryptage d'images

Introduction

L'information est un élément constitutif et déterminant dans tous les domaines, et sa sécurité est un domaine très vaste qui regroupe tous les aspects de la sauvegarde ou la protection des données. Tout au long de l'histoire, l'humanité a essayé d'envoyer des informations d'une façon sécurisée afin que leur contenu ne soit accessible qu'aux personnes ayant droit. A l'heure actuelle, les besoins en matière de sécurité d'informations sont grandissants, et la tendance n'est certainement pas à la baisse. En effet, de grandes quantités de données, en particulier les images numériques sont générées dans divers champs (militaire, médical, aéronautique, commerce,) et échangées via Internet et autres moyens de communication multimédia peu sécurisés. C'est dans la perspective d'assurer la protection de ces données secrètes contre les accès non autorisés que la cryptographie trouve sa raison d'être. La cryptographie a été longtemps utilisée comme moyen technique de chiffrement de données secrètes. Elle fait d'ailleurs l'objet de modernisation de nos jours en fonction de l'évolution de nouvelles technologies et constitue un sujet de recherche majeur dans les domaines académique et industriel.
Dans ce chapitre, il sera question pour nous de parcourir et d'apprécier l'évolution de la cryptographie depuis sa genèse jusqu'à nos jours et présenter quelques approches modernes sur le cryptage d'images fixes.
Nous commençons dans ce chapitre par une brève présentation de l'historique sur la cryptographie, sa terminologie ainsi que ses concepts de base. Ensuite, nous continuerons par la classification des systèmes cryptographiques, présenterons les modes de chiffrement et les généralités sur la fusion d'images. Nous clôturerons le chapitre par un état des contributions sur des crypto systèmes utilisant la fusion ou mixage d'images dans leur processus, puis présenterons les différents procédés de cryptanalyse.

1.1 Bref historique de la Cryptographie

La cryptographie est une Science très ancienne qui date de 1900 ans avant J-C. En effet, pour leurs différentes communications, les hommes ont développé plusieurs moyens (codes)

pour assurer la confidentialité de leurs échanges. Le mot cryptographie désigne un ensemble de techniques visant à rendre intelligibles des messages [64]. En parcourant l'histoire, on trouve des exemples de chiffres chez les Hébreux et les Grecs. Des recherches indiquent qu'un scribe égyptien a employé des hiéroglyphes non conformes à la langue pour écrire un message. De ce temps-là et au long de l'histoire, la cryptographie a été utilisée exclusivement à des fins militaires. Aujourd'hui, les réseaux informatiques exigent une phase de cryptographie comme mécanisme fondamental afin d'assurer la confidentialité de l'information numérique. Jusqu'aux systèmes contemporains, les codes étaient loin d'être aussi complexes et astucieux qu'à notre époque. La majeure partie des systèmes cryptographiques reposait sur deux principes fondamentaux : la substitution (consistant à remplacer certaines lettres par d'autres) et la transposition ou décalage (consistant à permuter des lettres du message afin de le brouiller). Le chiffre le plus simple consistait à remplacer chaque lettre du texte clair par un autre symbole (substitution simple), et ce symbole pouvait être une lettre du même alphabet, d'un autre alphabet ou un dessin.
Comme exemples de codes développés, en se rapportant à des événements historiques de notre ère informatisée, nous pouvons retenir les suivants ([65],[66]) : le bâton nommé « scytale » au 5e siècle avant JC, en passant par le carré de Polybe ou encore le code de César ; la version du code de César amélioré par Vigenère ; le système de Flayfair inventé par Sir Charles Wheatstone ; la machine Énigma créée par Dr Arthur Scherbius en 1923 et qui a été largement utilisée dans la seconde guerre mondiale ; la théorie de l'information de Claude Shannon en 1949 ; l'algorithme Lucifer développé par la société IBM en 1970 ; la théorie du système à clef publique par Whitfield Diffie et Martin Hellman en 1976 qui a marqué le début de la cryptographie moderne ; le système RSA, le plus utilisé actuellement des systèmes à clé publique inventé par Rivest, Shamir et Adleman en 1977 ; le premier standard pour le cryptage, l'algorithme DES en 1976 ; les résultats de la cryptographie quantique par Charles H. Bennett et Gilles Brassard en 1990, et le standard AES en 2000, qui a remplacé le DES.
En somme, en l'espace de quelques années, la cryptographie et la cryptanalyse sont passées de simples techniques désuètes, à de véritables sciences. Cette progression des techniques et algorithmes de cryptage ne s'est pas faite toute seule, c'est principalement à cause des attaques incessantes, visant à « casser » les techniques adverses, que l'on a pu assister à un tel bond.

1.2 Définition et terminologie de la cryptographie

La cryptographie et la cryptanalyse sont deux branches liées qui appartiennent à la cryptologie, science fondée sur les mathématiques qui étudie les communications sécrètes. La figure 1.1 illustre les différentes connexions entre elles.

- **La cryptographie** peut se définir comme l'étude des méthodes donnant la possibilité

d'envoyer des données de manière confidentielle sur un support donné. Ces données peuvent être de différents types (textes, images, vidéo, etc.).
D'une manière générale, nous appellerons la(les) donnée(s) à chiffrer texte en clair et celui obtenu après le processus de chiffrage texte chiffré.

- **La cryptanalyse** est l'étude des procédés cryptographiques dans le but de trouver des faiblesses et, en particulier, de pouvoir décrypter des textes chiffrés. En d'autres termes, c'est toute action consistant à retrouver le texte en clair sans connaître la clef de déchiffrement.

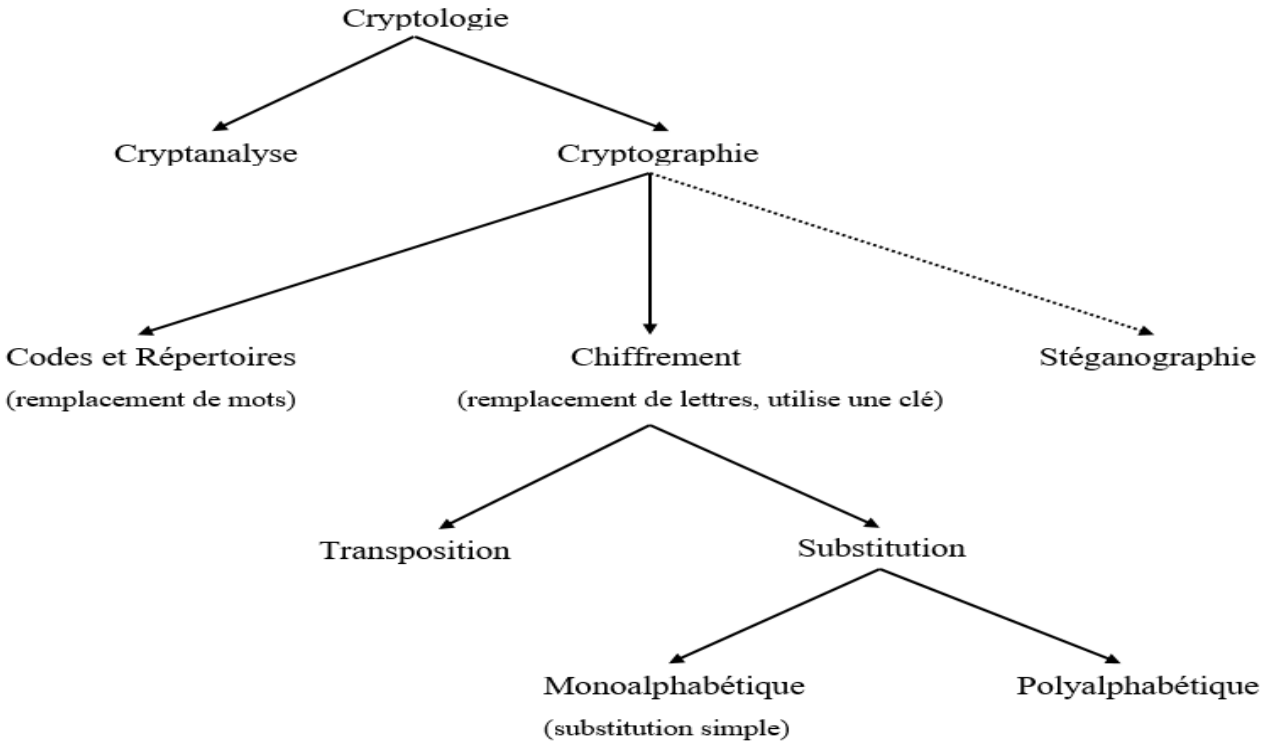


FIGURE 1.1 – Domaines inclus dans la cryptologie[67].

La cryptographie utilise des concepts issus de nombreux domaines (Informatique, Mathématiques, Electronique). Nous définissons quelques-uns pour une meilleure appropriation de ces différents concepts :

- **Chiffrement** : c'est l'opération qui consiste à transformer une donnée (texte, message, ...) afin de la rendre incompréhensible par une personne autre que celui qui a créé le message et celui qui en est le destinataire.
  La fonction permettant de retrouver le texte clair à partir du texte chiffré porte le nom de **déchiffrement**.
- **Crypter** : c'est rendre un message ou un texte en clair non intelligible pour des raisons de confidentialité.

- **Décrypter** : c'est l'action de retrouver le texte en clair correspondant à un texte chiffré sans posséder la clé qui a servi au chiffrement. Cette action est souvent opérée par des intrus au travers de différentes attaques (nous les détaillerons au dernier paragraphe de ce chapitre).
- **Texte en clair** *(plain text)* : c'est le message ou l'information à protéger.
- **Texte chiffré** *(Cipher text)* : Appelé également cryptogramme, le texte chiffré est le résultat de l'application d'un chiffrement à un texte clair.
- **Clef** : Il s'agit du paramètre impliqué et autorisant des opérations de chiffrement et/ou déchiffrement. C'est le secret partagé, utilisé pour transformer le texte en clair en texte chiffré et pour déchiffrer le texte chiffré en texte en clair.
  Dans le cas d'un algorithme symétrique, la clef est identique lors des deux opérations.
  Dans le cas d'algorithmes asymétriques, elle diffère pour les deux opérations. Il est important de noter qu'on peut construire un algorithme qui n'utilise pas de clef ; dans ce cas, c'est l'algorithme lui-même qui constitue la clef, et son principe ne doit donc en aucun cas être dévoilé.
- **Algorithme cryptographique** : c'est une fonction mathématique utilisée pour le chiffrement et le déchiffrement.
- **Attaque** : c'est toute action ou transformation malveillante appliquée à un cryptosystème visant à le décrypter le texte en clair.
- **Cryptosystème** : Il est défini comme l'ensemble des clés possibles (espace de clés), des textes clairs et chiffrés possibles associés à un algorithme donné.

Définition 1 : Un cryptosystème est un quintuplé (M, C, K, E, D) dans lequel M est appelé l'espace en texte clair ; c'est un espace appelé le texte chiffré ; K est appelé l'espace de clé ; E est une fonction de cryptage ; D est une fonction de décryptage. Pour chaque $k \in K$, il existe une fonction $e_k \in E$ et une fonction correspondante $d_k \in D$ tel que, pour chaque message en texte clair, $x \in P$, $d_k(e_k(x)) = x$. Notons que d_k est l'inverse de la fonction e_k.
Définition 2 : Une fonction $T : X \to Y$ est injective si $\forall a, b \in X$, $T(a) = T(b) \Rightarrow a = b$

1.3 Concepts de base de la cryptographie

1.3.1 Principes garantissant un bon cryptosystème

Le plus souvent, le canal par lequel l'Emetteur du message et le Récepteur communiquent est peu sécurisé, ce qui constitue une potentielle source d'attaque. De ce fait, un crypto système robuste devrait être bâti selon certaines règles pour remédier à cette problématique. Deux lois ont été élaborées à cet effet à savoir celle de Kerchoffs et de Shannon [68] :

— Une théorie fondamentale a été énoncée en 1883 par A. Kerckhoffs (cryptologue néerlandais). Elle suppose que l'intrus connaît tous les détails du cryptosystème sauf la clé.

Le secret doit entourer seulement la clé de cryptage. La clé doit être alors le sésame aboutissant à la solution [69]. La sécurité d'un système de chiffrement est beaucoup plus sûre si sa cryptanalyse fait un temps qui n'est pas meilleur que celui de la recherche exhaustive de la clé. Cette évaluation est appelée la sécurité calculatoire.

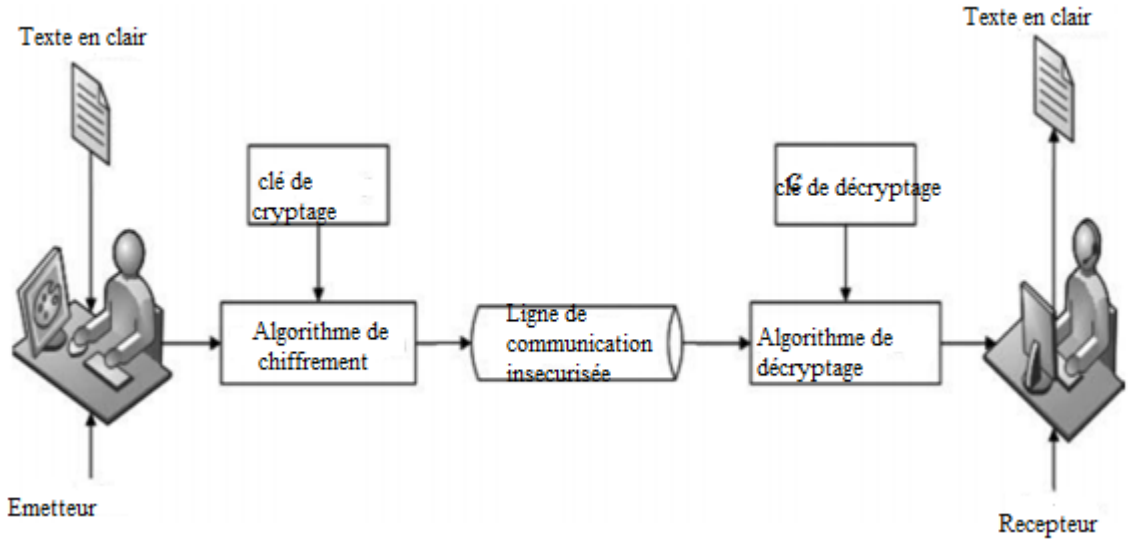


FIGURE 1.2 – Schéma block d'un processus de chiffrage/déchiffrage [70].

— - Shannon a énoncé dans ses travaux sur les fondements théoriques de la cryptographie [71] deux critères essentiels pour construire un chiffrement sécurisé pour obscurcir les redondances dans un message : ce sont la confusion et la diffusion. Un chiffrement qui vérifie ces deux propriétés s'avère difficile à être cassé [66].

La confusion est la relation complexe entre clé, message clair et message crypté. La méthode la plus simple pour appliquer la confusion est la substitution. Les exemples de chiffrement par substitution sont : Vigenère, César, Enigma.

La diffusion consiste à répartir la redondance du message clair et de la clé sur la plus grande longueur possible du message crypté. On peut avoir la propriété de diffusion par simple.

1.3.2 Objectifs de la cryptographie

La cryptographie repose sur quatre éléments essentiels : la confidentialité, l'intégrité des données, l'authentification et la non répudiation.

- **La confidentialité** : c'est le meilleur avantage qu'offre la cryptographie en matière de sécurité. Elle consiste à rendre l'information inintelligible à tous ceux qui pourraient intercepter les données ou message. La confidentialité peut être aussi vue comme la protection des données contre une divulgation non autorisée. Le secret de l'informa-

tion peut être accompli par différents moyens, en commençant par la sécurité physique jusqu'à l'utilisation d'outils scientifiques.

- **L'intégrité des données** : c'est un mécanisme qui permet de s'assurer que les données reçues par le récepteur n'ont pas été modifiées, altérées durant la transmission. Il s'agit d'un avantage en matière de sécurité qui reconnaît tout changement apporté à l'information, car l'information peut être modifiée par une personne non approuvée délibérément ou accidentellement. Ici, nous avons une idée claire sur la fiabilité de l'information qui a été échangée entre deux parties : si l'information a été contrôlée de manière approuvée ou non ; si l'information a été reçue dans sa totalité.
- **L'authentification** : il donne la preuve reconnaissable de l'initiateur du message. Il affirme au récepteur que l'information reçue a été envoyée par un l'expéditeur confirmé. L'avantage de vérification ou authentification comporte deux variantes :
  - **L'authentification du message** : consiste à reconnaître l'auteur du message sans tenir compte des contraintes du canal de transmission ;
  - **L'authentification de la personne/source** : consiste à s'assurer de l'affirmation selon laquelle l'information a été obtenue à partir d'un élément particulier, par exemple un site spécifique. Mis à part l'initiateur du message, la vérification peut également donner l'affirmation sur les différents paramètres de l'information telles que la date et l'heure de la création/transmission.
- **La non répudiation** : elle traduit la possibilité de vérifier que l'envoyeur et le destinataire sont bien les parties qui disent avoir respectivement envoyé ou reçu le message. Autrement dit, la non-répudiation ou le non-désaveu de l'origine prouve que les données ont été envoyées, et la non-répudiation de l'arrivée prouve qu'elles ont été reçues. Cette propriété est assez intéressante lorsqu'on se trouve dans une situation où il y a des chances de contradiction en ce qui concerne l'échange de données (dans le commerce électronique par exemple).

1.4 Classification des systèmes cryptographiques

Il existe plusieurs manières de classifier les algorithmes de cryptage. En général, les trois catégories suivantes sont utilisées :

— Cryptographie symétrique ou à clé secrète
— Cryptographie asymétrique ou à clé privée
— Fonctions de Hachage

Nous présenterons dans cette sous-section une brève description de ces groupes de systèmes cryptographiques, une explication plus détaillée peut être consultée dans les travaux de thèse de Abanda A. [72] portant sur les nouveaux crypto-systèmes basés sur le mixage et la fusion

des cartes de données.

1.4.1 Cryptographie symétrique

La cryptographie symétrique, également dite à clé secrète (par opposition à la cryptographie asymétrique), est la plus ancienne forme de chiffrement. Elle permet à la fois de chiffrer et de déchiffrer des messages à l'aide d'une même clé. L'émetteur et le récepteur utilisent la même clé qui doit être privée (figure 1.3) ou bien une clé peut être déduite de l'autre [73].

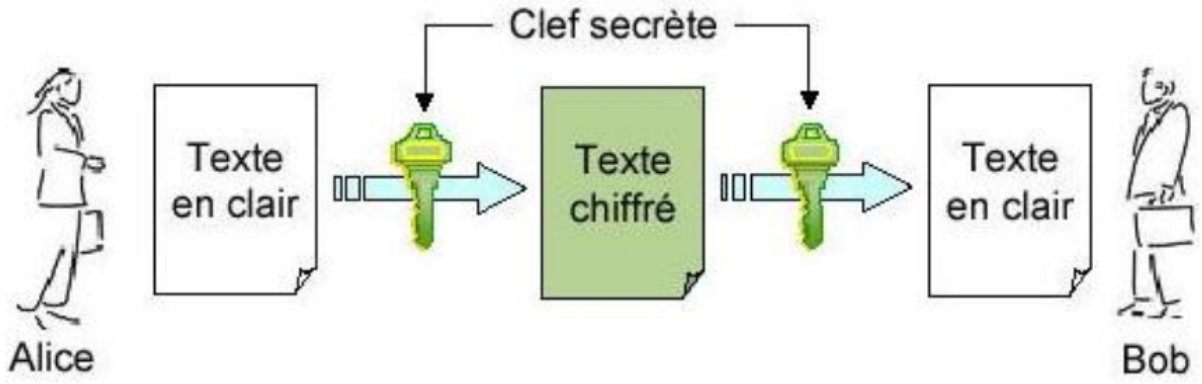


FIGURE 1.3 – Cryptographie symétrique. Figure tirée de [67].

Le chiffrement et le déchiffrement symétrique d'un message peuvent se faire de deux manières :

1.4.1.1 Chiffrement par flot (Continu)

Dans un chiffrement par flot (*Stream Cipher*), chaque bit est traité directement ; c'est-à-dire qu'on opère sur un flot continu de données. (Voir figure 1.4). Ce mode est adapté surtout pour la communication en temps réel et implémenté en général sur des supports hardwares. Il est plus facile à analyser.

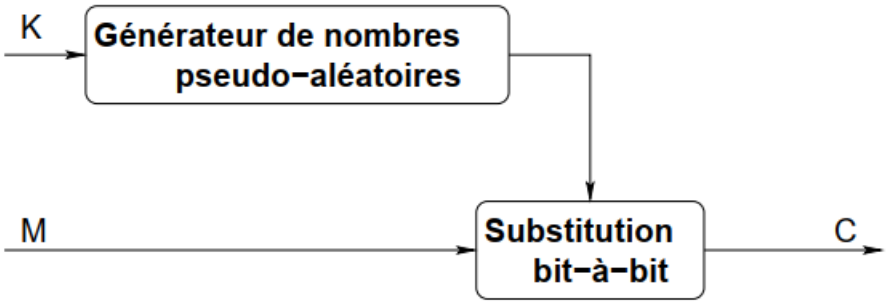


FIGURE 1.4 – Chiffrement symétrique par flot (Stream Cipher) [73].

1.4.1.2 Chiffrement par bloc

Dans le chiffrement par bloc (*Block Cipher*), chaque message est divisé en blocs de tailles fixes. On peut ajouter des bits néants à la fin du message pour obtenir des blocs entiers (généralement de 64 bits). Ce mode est adéquat pour l'implémentation logicielle en général. Ce chiffrement est plus répandu vu les performances logicielles des algorithmes. La sécurité augmente lorsque les blocs ont une longueur plus grande, mais la durée du traitement augmente alors notablement. Les exemples de cryptage symétrique par blocs sont multiples. On peut citer le DES et l'AES qui est le plus récent. La figure 1.5 illustre un chiffrement par bloc.

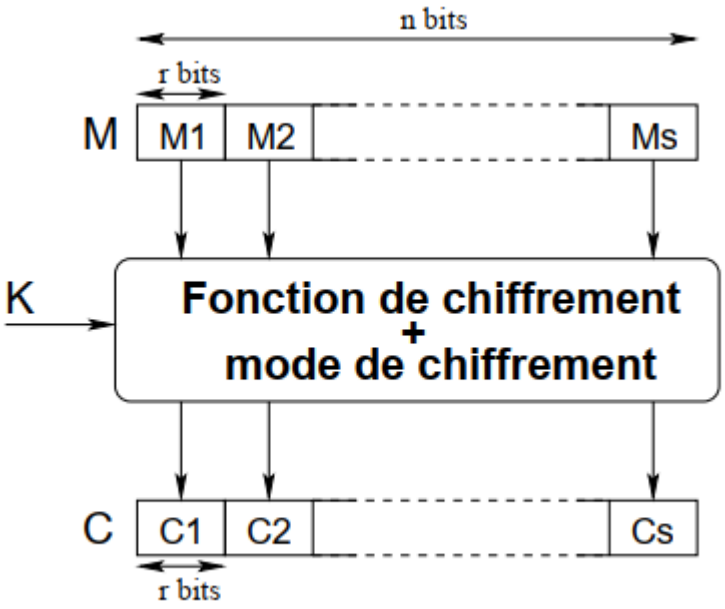


FIGURE 1.5 – Chiffrement symétrique par bloc (Block Cipher) [73].

Pour appliquer un chiffrement par bloc dans une variété d'applications, cinq modes de fonctionnement ont été définis par le NIST (SP 800-38A). Un mode de fonctionnement est une technique visant à accroitre l'effet d'un algorithme de cryptage ou l'adapter pour une application. Ces différents modes sont : ECB (Electronic Codebook), CBC (Cipher Block Chaining), et trois autres variantes issues des deux premiers modes : CFB (Cipher Feedback), OFB (Output Feedback), CTR (Counter) [68]. Ces modes sont décrits dans le tableau 1.1 et leurs schémas respectifs présentés dans la partie annexe du document.

TABLE 1.1 – Les cinq modes de chiffrement par bloc.

Mode	**Description**
Mode « Electronic Code Book » (ECB)	Il revient à chiffrer un bloc indépendamment des autres, cela permet entre autres de chiffrer suivant un ordre aléatoire (bases de données, etc.)
Mode « Cipher Block Chaining » (CBC)	Il permet d'introduire une complexité supplémentaire dans le processus de chiffrement en créant une dépendance entre des blocs successifs. Il effectue un XOR entre un bloc de données en clair et un bloc de données cryptées. Quant au premier bloc il est XORé avec un vecteur appelé vecteur d'initialisation (IV) qui peut être un mot de passe par exemple, ce vecteur change à chaque session, et doit être transmis au destinataire.
Mode « Cipher Feedback » (CFB)	Le message est ajouté par un XOR à la sortie du bloc chiffré. Le résultat sert d'entrée pour l'étape suivante. Il est utilisé pour le chiffrement par flux.
Mode « Output Feedback » (OFB)	Une variante du CFB. La différence ici, c'est que le flux entrant vers les étapes ultérieures est indépendant du message clair.
Mode « Compteur » (CTR)	Chaque bloc est XORé avec compteur chiffré qui incrémente chaque séquence de chiffrement.

Les algorithmes symétriques présentent aussi bien des avantages que des inconvénients qui sont résumés dans le tableau 1.2

TABLE 1.2 – Avantages et inconvénients du chiffrement symétrique

Avantages	**Inconvénients**
• Assure la confidentialité des données • Algorithme de cryptage performant • Plus utilisé pour la transmission de longs messages (débit plus important) • Les clés sont relativement de faible taille • Primitives de mécanismes cryptographiques	• Problème de distribution de clés : trouver un canal parfaitement sûr pour transmettre la clé • Problème de gestion de clés

1.4.2 Cryptographie asymétrique (à clé privée)

Le principe de chiffrement asymétrique (appelé aussi chiffrement à clés publiques) est apparu en 1976, avec la publication d'un ouvrage sur la cryptographie par Whitfield Diffie et Martin Hellman [74]. Dans un cryptosystème asymétrique, les clés existent par paires :

— Une clé publique pour le chiffrement ;

— Une clé secrète pour le déchiffrement.

La connaissance de la clé de chiffrement E_K ne permet pas de déduire celle de déchiffrement D_K.

Dans le procédé, il suffit de chiffrer le message à envoyer au moyen de la clé publique du destinataire (qu'il trouvera par exemple dans un serveur de clés). Ce dernier sera en mesure de déchiffrer le message à l'aide de sa clé privée (qu'il est seul à connaître). La figure 1.6 illustre le procédé de chiffrement asymétrique.

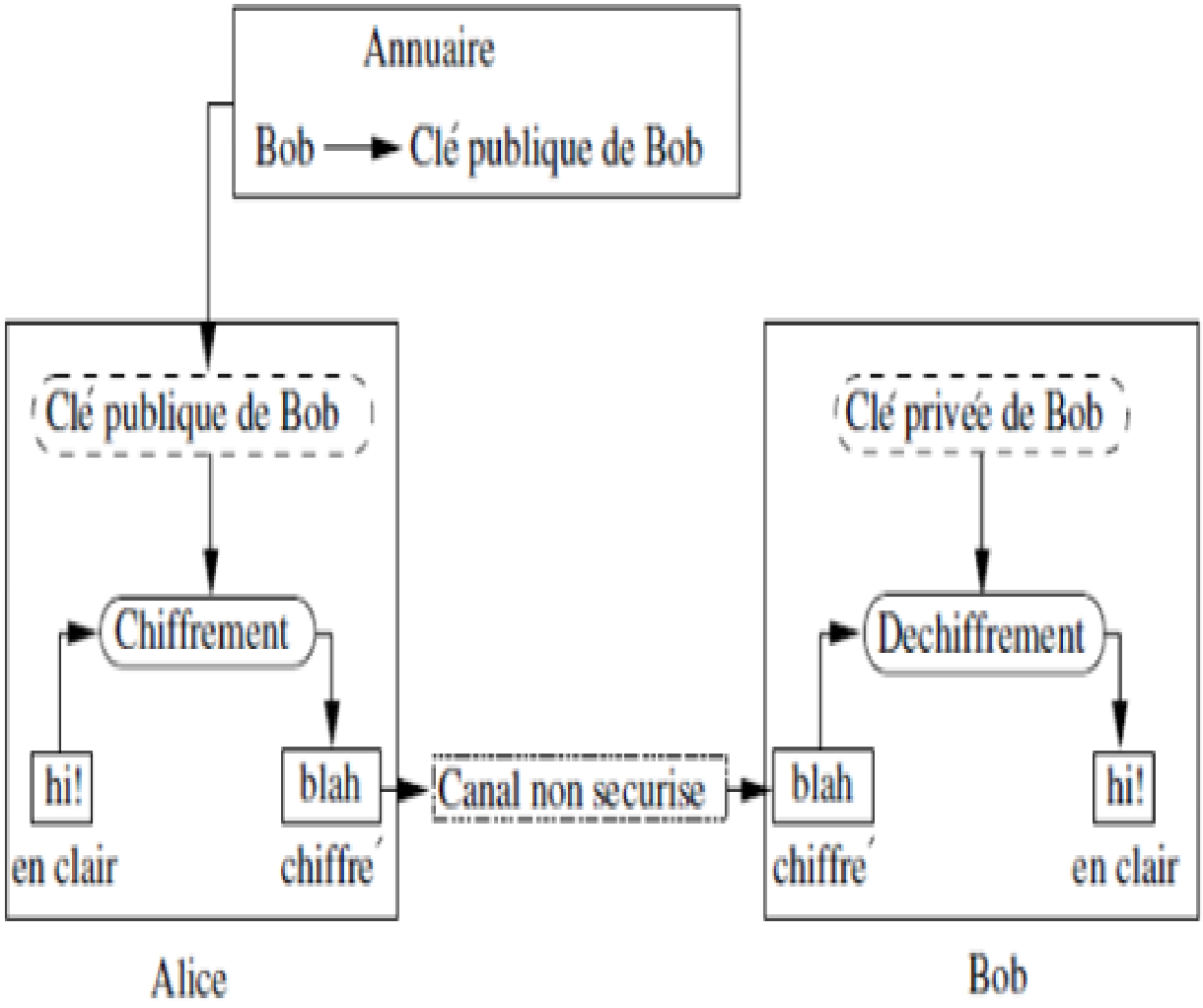


FIGURE 1.6 – Chiffrement asymétrique [75]

Les avantages et inconvénients du chiffrement asymétrique sont regroupés dans la table 1.3

TABLE 1.3 – Avantages et inconvénients du chiffrement asymétrique par rapport à celui symétrique

Avantages	**Inconvénients**
• La distribution des clés est simplifiée : la clé privée n'est jamais révélée ou transmise et la clé publique est disponible à tous les utilisateurs. • Certification des clés publiques par la signature numérique. • La paire de clés privée/publique reste inchangée pour une longue durée. • Le nombre de clés distribuées dans un large réseau est faible par rapport à celui d'une cryptographie symétrique.	• Visiblement plus lent que les algorithmes symétriques. • Garantir que la clé publique que l'on saisit est bien celle de la personne à qui l'on souhaite faire parvenir l'information cryptée : attaque d'usurpation d'identité. • La taille des clés est beaucoup plus importante que les clés symétriques.

1.4.3 Cryptographie Hybride

Les algorithmes à clé publique sont très lents, comparés à ceux à clé privée. Dans un souci de corriger ce manquement, des auteurs ont développé le chiffrement hybride qui fait appel aux deux techniques, symétrique et asymétrique, comme présenté sur la figure 1.7. Il a été mis en œuvre pour la première fois par Zimmermann pour le PGP (Pretty Good Privacy) en 1991 [68]. L'idée d'un système hybride est d'utiliser la rapidité de l'algorithme symétrique et la sécurité de l'asymétrique. Une clef secrète K de 128bits (ou plus) est générée automatiquement pour la session. Le message m est chiffré avec cette clef K en utilisant un chiffreur symétrique, $m' = e_k(m)$. La clef K est alors chiffrée avec un chiffreur asymétrique en utilisant la clef publique du destinataire B, $K' = e_{kB}^{Pub}(K)$. Ensuite, le message entier $M = m' + K'$ (message chiffré symétriquement et clef asymétriquement) est envoyé au destinataire. De l'autre côté, B utilise sa clef privée k_B^{Pri} pour décrypter la clef K' et ensuite déchiffrer le message. Un exemple d'un système hybride est le protocole SSL (Secure Socket Layer) développé par les sociétés Netscape et RSA Security.

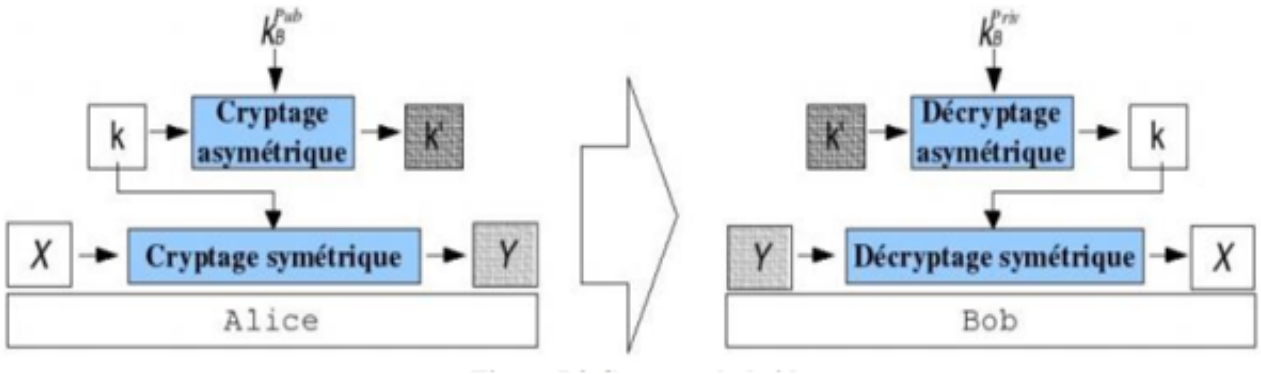


FIGURE 1.7 – Chiffrement hybride [75]

1.4.4 Fonctions de Hachage

Une fonction de hachage est une fonction permettant d'obtenir un condensé (ou haché) d'un texte (ou des images, vidéos, fichiers) en une suite de caractères assez courte représentant le texte qu'il condense. Une fonction de hachage doit être telle qu'elle associe un et un seul haché à un texte en clair (cela signifie que la moindre modification du document entraîne la modification de son haché). D'autre part, il doit s'agir d'une fonction à sens unique (one-way function) afin qu'il soit impossible de retrouver le message original à partir du condensé. S'il existe un moyen de retrouver le message en clair à partir du haché, la fonction de hachage est dite à brèche secrète.

Les fonctions de hachage sont très importantes dans plusieurs domaines notamment en sécurité informatique. Á la fin du 20^{e} siècle, plusieurs algorithmes de hachage les plus utilisés pour générer des empreintes ont été inventés. Il s'agit principalement de :

- **MD5** (en 1991) : Cette fonction de hachage très populaire produit des empreintes de 128 bits. Elle succède à MD4 dont elle améliore la sécurité, mais génère toutefois des empreintes trop courtes. Ainsi l'attaque des anniversaires a déjà été menée avec succès, d'où cette fonction n'est pas assez recommandée lorsqu'elle est utilisée seule dans un cryptosystème.
- **SHA-1** (en 1995) : Cette fonction conçue par la NSA (National Security Agency) produit des empreintes de 160 bits. Sa sécurité est réputée très bonne dans la mesure où de nombreuses études ont été réalisées à son sujet sans trouver de faille réelle. Cependant, très récemment, certains travaux ont permis de souligner quelques faiblesses la concernant. Ces faiblesses n'ont pas pour l'instant d'incidence pratique [73]. L'algorithme SHA-1 reste utilisé dans la plupart des applications cryptographiques.
- **SHA-256** (en 2000) : Cette fonction s'inspire du fonctionnement de SHA-1. Elle est

basée sur un chiffrement à clef secrète par bloc et utilise les blocs de b = 512 bits pour produire une empreinte de n = 256 bits (voir la figure 1.8 ci-dessous). La recherche de collisions par force brute nécessite 2128 calculs d'empreintes. A l'heure actuelle, il n'y a pas d'attaque efficace connue contre SHA-256 qui soit considée comme sûre.

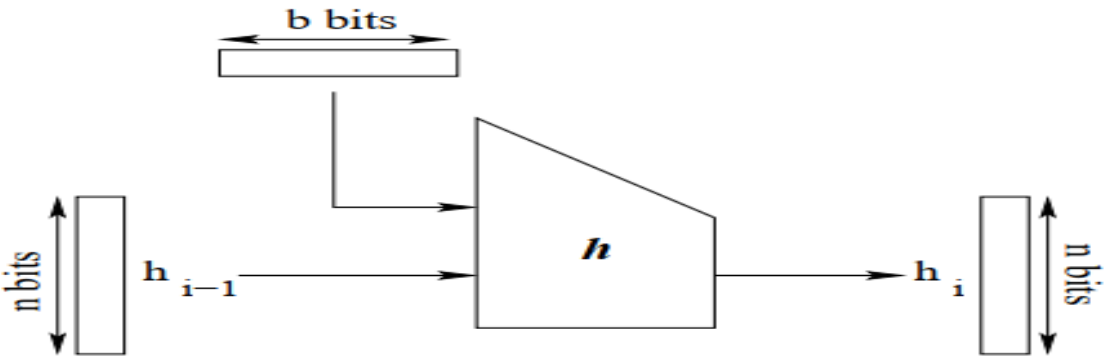


FIGURE 1.8 – Compression de bits par une fonction de hachage [73].

1.5 Outils de la cryptographie et quelques algorithmes de chiffrement modernes

1.5.1 Les générateurs de nombres aléatoires et pseudo-aléatoires

Il existe de nombreuses applications cryptographiques dans lesquelles la génération aléatoire des nombres, chaîne de bits, etc. est importante. Par exemple, les clés cryptographiques doivent être engendrées aléatoirement dans un certain espace de clés, et de nombreux protocoles utilisent des nombres aléatoires. Comme protocole, nous pouvons citer la génération de clés de session (clés utilisées une seule fois pour envoyer d'autres clés), les vecteurs d'initialisation (cas du mode CBC), des secrets nécessaires à la génération de signature (ElGamal), etc.
Un **générateur aléatoire** (en anglais, random generator) est un dispositif capable de générer des nombres de façon aléatoire, imprévisible (inattendue) et non reproductible. Un tel générateur est normalement doté d'un dispositif externe mesurant des phénomènes physiques connus par leur non déterminisme (par exemple une source radioactive ou quantique).
Une façon de générer des bits aléatoires est d'utiliser le hasard naturel qui se produit dans les phénomènes physiques de la nature [72]. Par exemple, le bruit thermique d'une résistance à semi-conducteur est connu pour être une bonne source de hasard. Cependant, produire des bits aléatoires ne serait pas pratique pour les applications cryptographiques, car la plupart des

conditions naturelles ne sont pas pratiques en raison de la lenteur inhérente dans l'échantillonnage du processus et la difficulté de s'assurer qu'un adversaire n'a pas observé le processus. La plupart des ordinateurs ont une méthode pour générer des nombres aléatoires qui sont facilement disponibles pour l'utilisateur. Par exemple, la bibliothèque C+ standard contient une fonction rand ( ) qui génère des nombres pseudo-aléatoires entre 0 et 65535.
Dans les cryptosystèmes actuels, on utilise de moins en moins des générateurs aléatoires qui sont en général des boites noires et dont la connaissance du principe général de fonctionnement du générateur peut suffire pour identifier les paramètres de celui-ci à partir d'un échantillon des nombres qu'il produit. Les générateurs aléatoires produits par les ordinateurs sont périodiques et prévisibles sur la durée, comme substitut on utilise plus des générateurs pseudo-aléatoires car ils ont la particularité d'être déterministes tout en respectant certaines propriétés statistiques et certains critères de qualité attribués aux nombres aléatoires [72].
Les **générateurs pseudo-aléatoires** sont des procédés déterministes développés à partir d'une séquence aléatoire initiale (seed) pouvant être obtenue par des méthodes diverses (la fréquence de frappe d'un utilisateur, le nombre d'accès disque, le nombre de paquets reçus par une interface réseau, etc.). La fonction pseudo-aléatoire prend une graine comme entrée et produit une série de bits en sortie.
Il existe deux grandes familles de générateurs pseudo-aléatoires : les générateurs engendrés par les procédés algorithmiques et les générateurs physiques. Une étude détaillée de ces générateurs peut être consultée dans les travaux de Abanda A.[72]

1.5.2 Cryptage par fusion ou mixage d'images : état de l'art des contributions

1.5.2.1 Généralités sur la fusion de données/d'images

La fusion de données est la discipline qui cherche à combiner des informations obtenues de différents systèmes dans le but d'effectuer des inférences à partir de ces observations. Elle est un domaine en émergence avec des applications variées telles que la surveillance des océans, la défense aérienne, le renseignement sur les champs de bataille et l'identification de cibles, l'acquisition de données, le contrôle automatisé, la surveillance d'équipements complexes et la robotique.

- **Définition et objectif de la fusion**

Il existe plusieurs définitions de la fusion en fonction du domaine d'application. On définit généralement la fusion d'images comme la combinaison de deux ou de plusieurs images différentes pour former à l'aide d'un algorithme une nouvelle image [76]. En télédétection par

exemple, la fusion d'images consiste à produire une nouvelle image qui conserve l'information contenue dans chacune des images originales. Ici, l'objectif visé est de créer une synergie, c'est-à-dire d'obtenir une image fusionnée géométriquement et/ou sémantiquement plus riche qu'une image initiale. Pour les applications telles que le cryptage d'images, la fusion ou le mixage vise à obtenir une image finale dégradée, décorrélée et dont les détails de texture de chacune des images initiales ne s'identifient pas visuellement.

- **Intérêts de la fusion d'images**

— Amélioration de la résolution spatiale de l'image composite ;
— Combinaisons d'informations diachroniques ;
— Réduction du nombre d'images à traiter ou transmettre ;
— Transmission simultanée de plusieurs images ;

- **Niveaux de fusion d'images** [76]

La fusion peut s'effectuer à trois moments différents du travail (Figure 1.9) : soit au niveau du pixel (fusion des pixels), soit au niveau des caractéristiques après une segmentation (fusion des objets extraits de l'image), soit au niveau décisionnel lors de la phase finale de la segmentation (fusion des objets extraits et identifiés) [76].

- Pour effectuer la fusion au niveau du pixel, les capteurs d'images doivent être identiques (par exemple, plusieurs caméras infrarouges) ou commensurables (par exemple, images infrarouges et images radar).
- Pour la fusion au niveau des caractéristiques, un vecteur d'attributs est extrait à partir de la sortie de chaque capteur. Ces vecteurs d'attributs sont ensuite combinés (fusionnés) et une déclaration d'identification est ensuite effectuée sur la base de ce vecteur conjoint. Les outils utilisés pour la déclaration d'identification comprennent les techniques statistiques telles que l'analyse de regroupements, les réseaux de neurones, les techniques structurelles et à base de connaissances.
- Dans la fusion au niveau des décisions (ou niveau des déclarations), chaque capteur effectue de façon indépendante un estimé ou une déclaration de la scène observée. Ces estimations sont ensuite combinées à travers un procédé de fusion. Les techniques de fusion de déclarations incluent les méthodes de votation, l'inférence bayésienne et la théorie de Dempster-Shafer (Théorie de l'évidence).

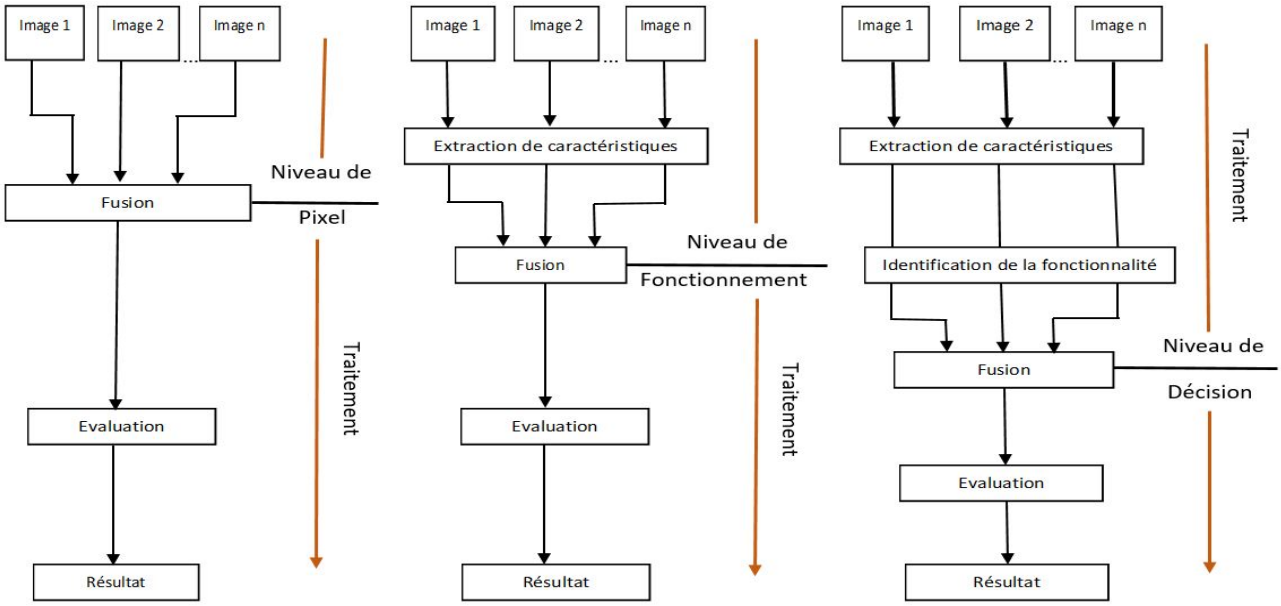


FIGURE 1.9 – Niveaux de fusion d'images [76].

Dans le cadre de notre travail, nous ferons une fusion d'images au niveau du pixel en vue d'obtenir une image chiffrée à partir de laquelle les images initiales peuvent être reconstruites sans perte d'informations significatives.

Pour une fusion au niveau du pixel, quelques conditions génériques doivent être observées :

- L'image fusionnée doit préserver toutes les informations pertinentes contenues dans les images sources aussi étroitement que possible ;
- Le processus de fusion ainsi qu'aucune autre étape ultérieure de traitement d'images ne doivent introduire des objets indésirables ou artefacts, qui peuvent tromper l'observateur humain ;
- Dans l'image fusionnée, les caractéristiques non pertinentes et les bruits doivent être supprimés à un degré maximum.

• **Champs d'application de la fusion d'images**

Les champs d'application de la fusion d'images sont variés :

- L'aide à la navigation :

Dans de mauvaises conditions météorologiques, les pilotes sont confrontés aux difficultés de visibilité telles que le brouillard ou la forte pluie. Ainsi, les avions et hélicoptères sont équipés de capteurs d'imagerie qui peuvent être consultés par le pilote. Parmi ces capteurs, le pilote dispose d'un capteur de lumière basse et un capteur infrarouge pour les images thermiques qu'il peut observer dans son écran d'affichage. Une amélioration possible est de combiner les deux sources d'images dans une seule image fusionnée qui contient l'information appropriée

des deux dispositifs imageurs.

- Imagerie médicale

Plusieurs techniques d'imagerie dans le diagnostic médical sont développées, et il se pose la nécessité d'une combinaison de tous les ensembles de données d'images disponibles afin d'obtenir une meilleure qualité de l'image finale.

- Télédétection

La fusion d'images est beaucoup utilisée dans le domaine de télédétection : de grandes bandes spectrales sont recueillies dans l'espace par des modules modernes à balayage spectral. Ces différentes bandes spectrales peuvent être visualisées et traitées individuellement, ou peuvent être fusionnées en une seule image, en fonction de la tâche d'analyse d'image. La nécessité de fusionner les images peut se présenter lors de différentes applications :

- Établir et mettre à jour une carte topographique.
- Étudier les occupations du sol.
- Aider à la décision dans l'agriculture, la foresterie et la faune.
- Surveiller les catastrophes naturelles, les neiges et les glaces.
- Améliorer les capacités d'interprétation et de lisibilité.

- Cryptographie

Afin de transmettre de grandes quantités de données, par exemple les images médicales, il est important de les protéger contre les personnes non autorisées, et une approche intéressante pour le faire est la fusion d'images cibles en une seule. En observant la structure de l'image fusionnée, cette dernière sera fortement décorrélée et ne laissera pas transparaitre à l'observation les détails de chacune des images utilisées. En plus, l'emploi de la fusion en cryptographie permet de réduire la quantité de données à transmettre de manière sécurisée sans une nette dégradation de ces dernières.

- Etc.

- **Quelques méthodes de fusion d'images** [77]

Plusieurs méthodes de fusion ou mixage d'images ont été proposées dans la littérature. Les méthodes de fusion d'images existantes peuvent être classifiées dans plusieurs groupes : les techniques du domaine spectral ; les techniques du domaine spatial et celles de l'espace échelle [78] .

Entre autres techniques, nous pouvons citer : la technique IHS (Intensity Hue Saturation) ; la transformation de Brovey, l'analyse en composantes principales, l'analyse multi résolution (transformations pyramidales), la transformation en ondelettes, la décomposition empirique, la transformation discrète en cosinus (DCT), les réseaux de neurones artificiels, le multiplexage spectral (par transformations optiques et générateurs chaotiques) , les techniques de filtrage, etc. certaines approches utilisent aussi une combinaison de quelques techniques citées précé-

demment.

- La transformation IHS (Intensity, Hue, Saturation)

L'intensité, la texture et la saturation sont les trois propriétés d'une couleur qui donnent une représentation visuelle contrôlée d'une image. La méthode de transformation IHS est la plus ancienne méthode de fusion d'images. Dans l'espace IHS, la texture et la saturation doivent être contrôlées parce qu'elles contiennent la plupart des informations spectrales. Selon cette technique, une image cible (panchromatique) de haute résolution et une ou plusieurs images multispectrales sont fusionnées en ajoutant des informations détaillées de haute résolution spatiale aux images spectrales. L'information fusionnée est donc de haute résolution. La figure 1.10 présente un exemple de fusion d'images panchromatique et multispectrales par la transformation IHS.

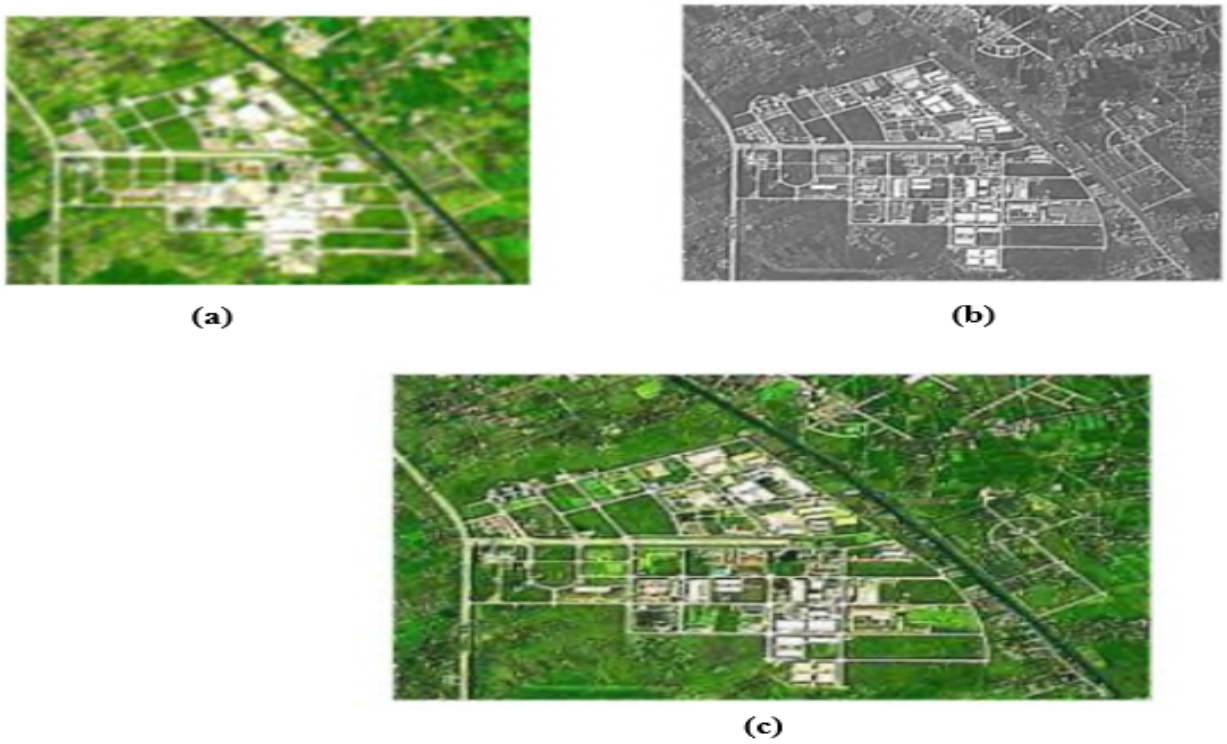


FIGURE 1.10 – Fusion d'images par la transformation IHS. (a) Image Multispectrale, (b) Image Plan, (c) Image fusionnée [77].

- L'analyse en composantes principales (Principal Component Analysis)

La méthode d'Analyse en Composantes Principales (PCA) est similaire à la transformation IHS, mais elle présente l'avantage d'utiliser un nombre arbitraire de bandes multispectrales.

C'est l'une des méthodes les plus populaires pour la fusion d'images. Plusieurs composants principaux non corrélés sont formés à partir des images multispectrales à faible résolution. Le premier composant principal (PC1) a l'information qui est commune à toutes les bandes d'images utilisées. Il contient une grande variance de telle sorte qu'il donne plus d'informations sur l'image panchromatique. Dans le processus de fusion, une image panchromatique de haute résolution est étirée pour avoir la même variance que PC1 et remplace PC1. Puis l'inverse de ta transformation PCA est effectuée pour obtenir l'image multispectrale de haute résolution [79]. Le deuxième élément principal (PC2) est fait perpendiculairement au premier et au troisième dans le sous-espace.

- Transformation en Ondelettes (Wavelet Transform)

La transformée en Ondelettes est considérée comme une alternative de temps très court par rapport à la transformée de Fourier. Elle est avantageuse à cette dernière parce qu'elle fournit la résolution souhaitée dans le domaine temporel ainsi que dans le domaine fréquentiel tandis que la transformée de Fourier donne une bonne résolution dans le domaine de fréquence seulement. Dans la fusion d'images par la transformation en Ondelettes, les images d'entrée sont décomposées en coefficients d'informations et d'approximation à différents niveaux spécifiques. Une règle de fusion est appliquée pour combiner ces deux coefficients et l'image qui en résulte est obtenue en prenant la transformation inverse d'ondelettes [80].

- Le filtre passe haut

Les images multispectrales haute résolution sont obtenues à partir d'un filtre passe haut. L'information à haute fréquence de l'image panchromatique haute résolution est ajoutée à l'image multispectrale de basse résolution pour obtenir l'image résultante. L'opération de fusion peut être réalisée soit en filtrant l'image panchromatique haute résolution avec un filtre passe haut, soit en prenant l'image originale panchromatique haute résolution et en soustrayant l'image faible résolution d'elle.

- Transformations pyramidales

Les pyramides d'images peuvent être décrites comme un modèle de fusion binoculaire pour le système visuel humain. En formant la pyramide structurée, une image originale est représentée à différents niveaux. Une image composite est formée en appliquant un modèle approche sélective de la fusion d'images. Tout d'abord, la décomposition pyramidale est effectuée sur chaque image source. Toutes ces images sont intégrées pour former une image composite, puis la transformation pyramidale inverse est appliquée pour obtenir l'image fusionnée.

- Les réseaux de neurones artificiels

Les réseaux neuronaux artificiels (ANN) ont trouvé leur importance dans la reconnaissance des motifs. En cela, une réponse représentée par une fonction non linéaire est utilisée. Il utilise un réseau d'impulsions de neurones couplés (PCNN) constitué d'un réseau de rétroaction. Ce réseau est divisé en trois parties, à savoir le champ réceptif, le champ de modulation et le

générateur d'impulsions. Chaque neurone correspond au pixel de l'image d'entrée. L'intensité du pixel correspondant est utilisée comme entrée externe au PCNN. Cette méthode lorsqu'elle est utilisée pour la fusion d'images est avantageuse, car résiste contre le bruit et comble les variations d'intensités dans les images cibles à fusionner [81] .
Parmi les techniques de fusion évoquées plus haut, la plupart n'est pas adaptée au cryptage d'images, car bon nombre d'entre elles ne sont pas réversibles ; pour celles qui le sont (DCT, transformation en Ondelettes), elles ne peuvent pas bâtir toutes seules un crypto-système robuste pouvant résister aux principales attaques. Par la suite, nous nous intéresserons plus aux techniques qui sont adaptées aux applications de cryptage d'images, notamment les images médicales.

1.5.2.2 État des contributions du cryptage par fusion/mixage d'images

Par le passé, la grande majorité des systèmes de fusion d'images était dédiée aux applications de prises de décision (reconnaissance de forme, classification, contrôle automatisé, etc.). En parcourant la littérature, des auteurs ont proposé ces années récentes des algorithmes de cryptage basés sur la fusion d'images. Certains d'entre eux utilisent la fusion comme une étape dans le processus de chiffrement ([55], [51], [59], [52]). Ces algorithmes utilisant la fusion d'images dans le processus de chiffrement peuvent se grouper en trois grandes catégories : les techniques utilisant le multiplexage spectral, les techniques de crypto-compression et celles respectant l'architecture schéma permutation-diffusion.

1. Méthodes utilisant le multiplexage spectral
Dans la méthode de fusion par multiplexage spectral, les images sont combinées en une seule dans le domaine spectral suite à une ou plusieurs transformation(s) mathématiques réversibles. Les transformations les plus utilisées sont : la transformée de Fourier avec sa variante Fractionnaire, la transformation discrète en cosinus ; la transformation en ondelettes. Entre autres algorithmes de ce groupe,

Isha et al.[51] ont proposé un algorithme de cryptage par fusion d'images basé sur la transformation en ondelettes pour sécuriser des images. Dans ce crypto-système, le processus de génération de clés est asymétrique. Les images originales sont codées en deux masques de phase, utilisant le principe d'interférence, puis ces images sont fusionnées par la transformation en ondelettes qui combine les composantes hautes et basses fréquences respectivement des différentes images. L'algorithme proposé permet de crypter plusieurs images de différents types avec une bonne qualité d'images reconstruites, sauf que les auteurs n'ont pas renseigné sur la valeur des principales métriques qui permettent d'évaluer la robustesse du système conte les attaques.

Shi D. et al. [52] ont proposé une nouvelle technique de fusion et cryptage simultanés d'images utilisant le concept de détection et compression. Au cours de leur analyse expérimentale, ils ont utilisé des modèles de multiplexeurs codés pour éclairer plusieurs objets simultanément. La lumière mixte réfléchie par les objets est reçue par un détecteur d'un seul pixel, et une méthode de reconstruction itérative a été utilisée pour restaurer l'image fusionnée en résumant les motifs multiplexés et les intensités détectées. La figure 1.11 présente le résumé des étapes de chiffrage adapté au modèle expérimental utilisé.

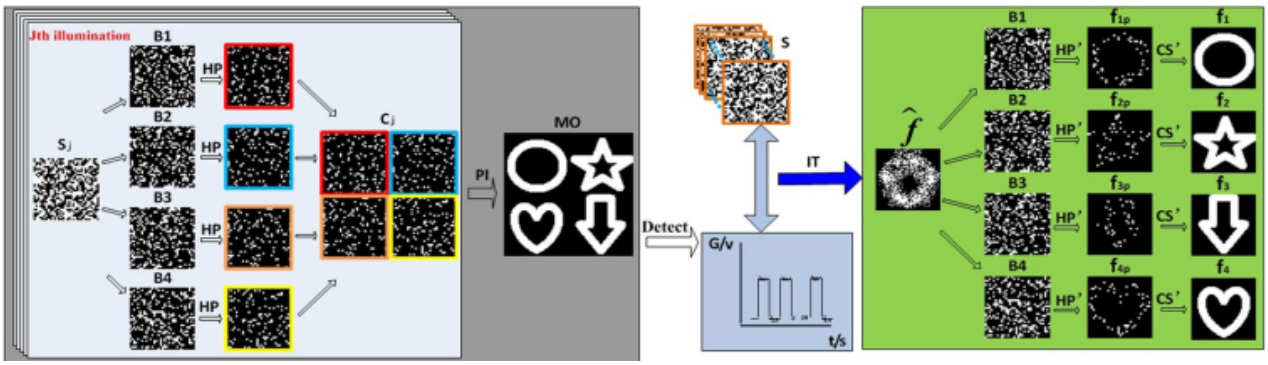


FIGURE 1.11 – Procédure de la méthode de chiffrement proposée par Shi D. et al.[52].

HP : produit de Hadamard ; **Bj** : matrices codées en binaire ; **Cj** : image fusionnée ; **Sj** : modèle de l'image multiplexée(fusionnée) ; **IT** : méthode de reconstruction itérative pour retrouver l'image fusionnée ; **HP'** : produit d'Hadamard entre l'image fusionnée $\hat{f}$ et les matrices codées **Bj** ; **f1p** , **f2p**, **f3p**,**f4p** : quatre échantillons d'images aléatoires ; CS' : algorithme de détection compression utilisé pour retrouver les quatre images cibles f1, f2, f3 et f4.
Tout d'abord, quatre matrices complémentaires B1, B2, B3 et B4 codées en binaire sont générées ainsi que le modèle de la matrice multiplexée S. Ensuite, la matrice fusionnée C est obtenue en faisant le produit de chacune des images cibles par les matrices codées en binaire : ici la transformation utilisée pour le multiplexage est le produit de Hadamard. Les images cibles peuvent être restaurées à l'aide d'une méthode de reconstruction itérative. La technique proposée offre de bonnes performances en terme de robustesse tant que le nombre d'images multiplexées augmente, mais elle présente des faiblesses au niveau de l'espace de clé réduit, et de la qualité limitée des images reconstruites.

Une autre technique de fusion spectrale découlant d'une amélioration de la transformée de Fourier Fractionnelle a été proposée par Zhengjum et al. [82]. Les auteurs réalisent un

double cryptage basé sur l'utilisation de différents masques et de la transformée de Fourier fractionnelle. Le schéma de principe de la transformée de Fourier Fractionnelle est présenté à la figure 1.12. La technique ainsi proposée permet de crypter, simultanément et avec un mode itératif, deux images en une seule image/amplitude. Afin d'augmenter les performances en termes de cryptage dans leur approche, ils proposent de rajouter dans le processus l'utilisation des phases aléatoires associées aux différentes images à crypter. Pour ce faire, ils regroupent les deux images initiales dans le domaine de Fourier fractionnel. À partir de l'image cryptée et de sa phase, il est possible d'obtenir séparément les deux images originales en utilisant la transformée de Fourier fractionnelle avec deux ordres α et β. L'algorithme itératif de cryptage est détaillé en figure 1.13. Avec F^{α} qui définit la transformation de Fourier fractionnelle d'ordre α ; A1, A2 sont les deux images originales à crypter. ϕ_i est la fonction de phases aléatoires associée à chacune des images cibles (images à crypter). Tous les paramètres utilisés dans l'élaboration de cet algorithme peuvent être considérés comme des clés de cryptage supplémentaires.

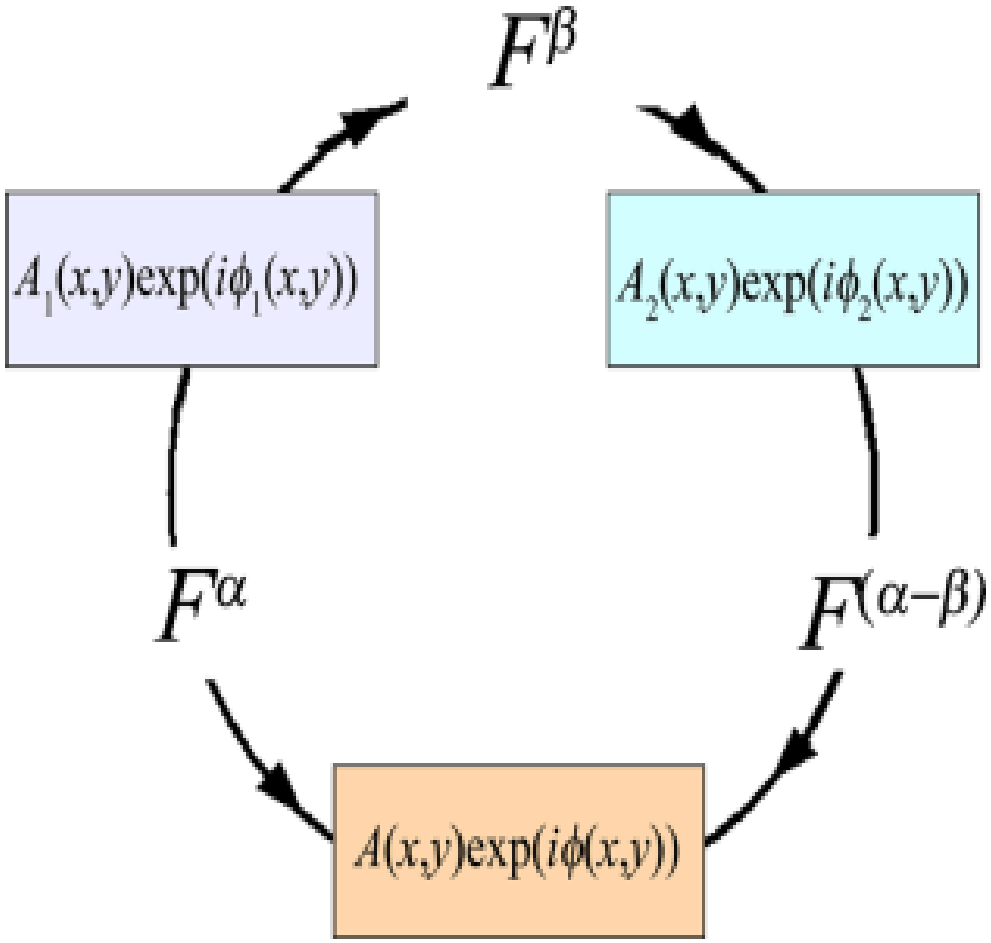


FIGURE 1.12 – Synoptique du cryptage basé sur la Transformée de Fourier Fractionnelle [82].

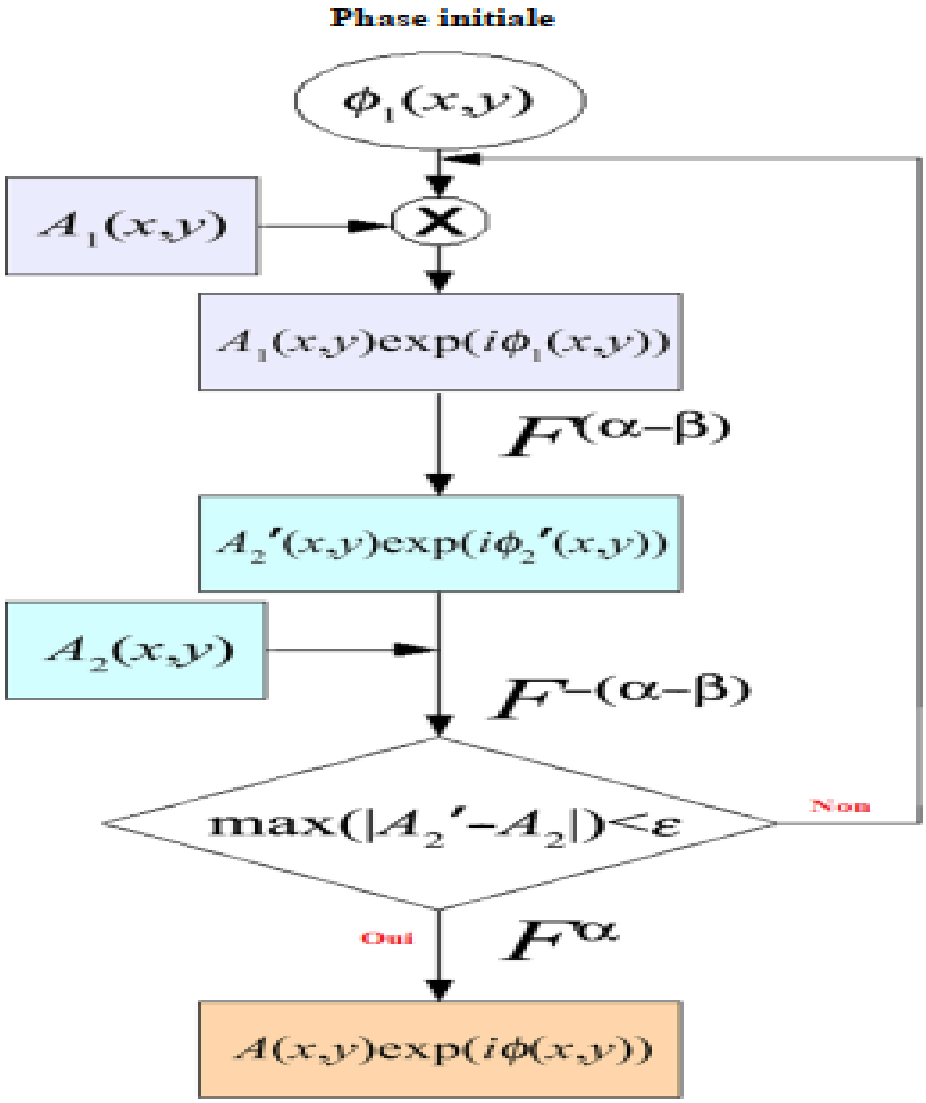


FIGURE 1.13 – Algorithme itératif de cryptage [82].

Les deux images originales A1 et A2 doivent satisfaire la condition suivante :

$$F^{\alpha}(A_1 exp(i\varphi_1)) = F^{\beta}(A_2 exp(i\varphi_2)) \tag{1.1}$$

Pour obtenir les deux phases φ_1 et $\varphi_2{}'$, l'algorithme itératif de la figure 1.13 doit être utilisé. Il consiste à fixer la valeur de φ_1 et de chercher la valeur de φ_2 qui minimise $max|A_2{}' - A_2|$ où

$$A_2{}'(exp(i\varphi_2{}')) = F^{\alpha-\beta}(A_1 exp(i\varphi_1)) \tag{1.2}$$

Cette méthode est interessante, car elle permet de faire une généralisation du multiplexage de plus de 2 images par la transformée de Fourier Fractionnaire à différents ordres. Cependant, elle présente des limites au niveau du temps d'implémentation qui est assez coûteux lorsque le nombre d'images à multiplexer augmente. En plus, les auteurs n'ont pas pris en compte l'effet des différents paramètres du cryptosystème sur la qualité de l'image décryptée, car certains de ces paramètres dégradent nettement l'image reconstruite. Une optimisation des performances de la technique proposée peut être faite afin de résoudre les limitations sus-évoquées.

Ayman A. et A. Mansour [63] ont développé une nouvelle approche de cryptage basée sur l'utilisation des méthodes de cryptage par filtrage fréquentiel et analyse des composantes indépendantes « ICA » (Independent Component Analyses). Cette technique permet à la fois de crypter l'information et le canal de transmission. Les auteurs utilisent une séquence vidéo de trois images de taille $(N \times N)$ pixels pour illustrer leur technique (voir figure 1.14-a).
La première étape de cette technique consiste à crypter les différentes images de la séquence avec une des techniques optiques (figure 1.14-b). Ensuite, ils mixent ensemble ces différentes images cryptées (figure 1.14-c) en utilisant un mixeur linéaire ($Mi \times 1 = a11D1 + a12D2 + a13D3$: avec D1, D2, D3 les trois images cryptées de manière optique et a11, a12, a13 les différents paramètres de mixage). Ce mixage linéaire va donner trois autres images cryptées. Ainsi nous obtenons une série d'images cryptées deux fois en utilisant des clés différentes et deux méthodes de cryptages différentes, ce qui a pour but d'augmenter le niveau de cryptage de ces images (figure 1.14-d). Avant de transmettre ces différentes images, considérées comme trois matrices, de $(N \times N)$ pixels chacune, et afin d'augmenter davantage le niveau de sécurisation, ils les convertissent en un seul vecteur ligne de taille $(3 \times N^2)$. Ensuite, ils changent l'ordre des différents pixels que constitue ce vecteur en utilisant un critère bien défini (ce critère sera utilisé comme clé supplémentaire de cryptage). Ensuite ce vecteur sera divisé en trois vecteurs et envoyé séparément sur trois canaux différents figure (1.14-e). Ainsi, quiconque intercepte un de trois messages ne pourra pas remonter à la source et trouver les informations en clair. Pour décrypter l'information transmise, nous devons réaliser les étapes inverses en utilisant les différentes clés de cryptage utilisées par l'émetteur pour crypter la séquence en question (figure 1.14-f) ainsi qu'utiliser la méthode ICA pour retrouver les trois images mixées [63], [83]). Malgré les atouts que présentent cette technique, elle a la faiblesse de causer les pertes d'informations au niveau des images décryptées, ceci est dû à l'utilisation de la technique ICA. Pour une amélioration de cette méthode de chiffrement, un bon compromis peut être trouvé entre la robustesse de l'algorithme contre les principales attaques et la qualité des images reconstruites.

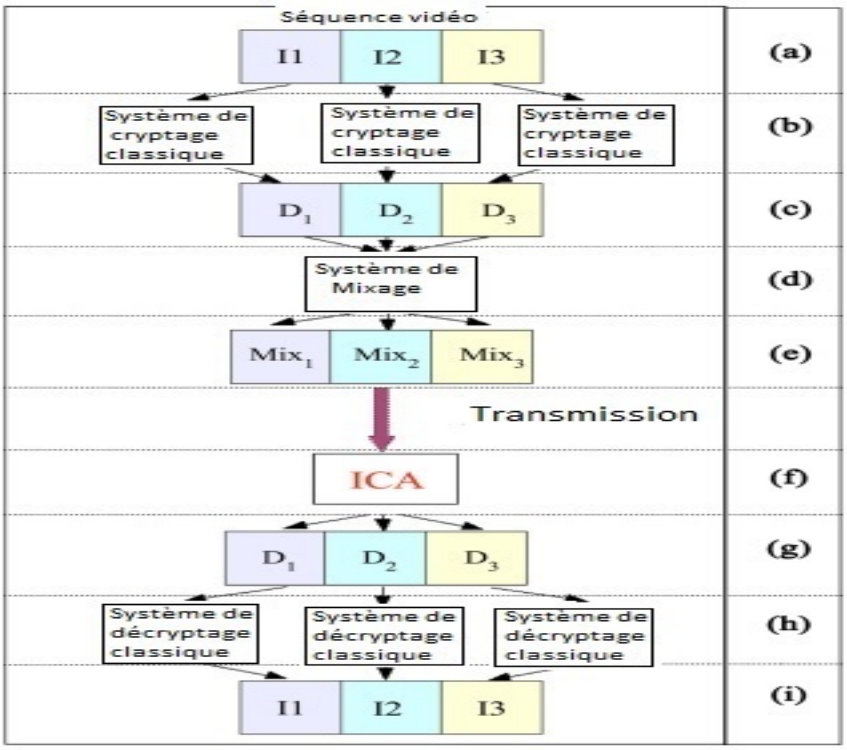


FIGURE 1.14 – Synoptique d'un système de cryptage/décryptage utilisant les techniques ICA [63].

Les mêmes auteurs, Ayman A. et A. Mansour [53] ont proposé une nouvelle technique de cryptage à deux niveaux de sécurité pour fusionner et crypter simultanément plusieurs images. Au regard des faiblesses que présente la méthode de cryptage double phase aléatoire (en anglais DRPE : Double Random Phase Encoding), Ayman et Mansour se sont proposés de l'améliorer et la généraliser pour le chiffrage de plusieurs images tout en réduisant le nombre d'images à transmettre. Le processus de chiffrement comprend deux phases : dans un premier temps, les images cibles sont fusionnées après application d'un ensemble de transformations itératives de Fourier. Cette fusion par multiplexage est réalisée selon la relation suivante,

$$I_j e^{i\varphi_j} = FT(I_{j-1} e^{i\varphi_{j-1}}) \tag{1.3}$$

où φ_j est la phase de l'image cible j et φ_{j-1} est la phase de l'image cible précédente. A ce niveau, plusieurs clés de codage, y compris des images biométriques sont utilisées. Par la suite, l'image mixte obtenue constitue l'entrée du système de cryptage par la technique DRPE classique. Dans l'approche proposée, le mixage des différentes images cibles donne lieu à une image unique qui contient toutes les informations nécessaires pour le processus de décryptage. Le schéma détaillé de la technique de cryptage proposée est indiqué à la figure 1.15. Les résultats obtenus après

simulations montrent de bonnes performances en terme de robustesse. Seulement, lorsque le rapport de compression dépasse 75%, la qualité des images reconstruites est fortement dégradée (avec un MSE autour de 4.94×10^{-1} pour une image de taille 256×256). Les auteurs ont pris en compte cette faiblesse et entendent y apporter des améliorations dans leurs prochains travaux.

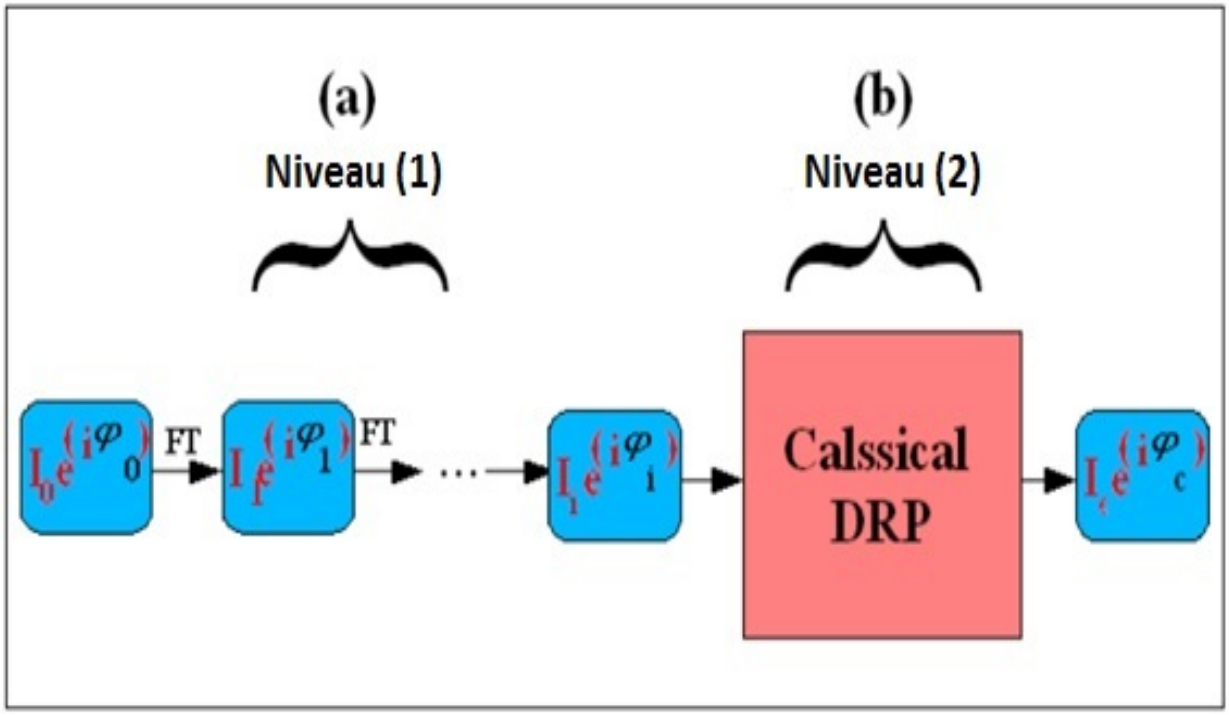


FIGURE 1.15 – Diagramme synoptique de la technique proposée. Figure tirée de Ayman et Mansour (2009).

2. Méthodes basées sur la crypto-compression

Plusieurs auteurs se sont intéressé ces dernières années aux algorithmes basés sur la crypto-compression, qui présentent l'avantage d'avoir un bon compromis compression/cryptage. Ce type de technique permet de transmettre de manière simultanée un ensemble d'images qui sont soit compressées puis cryptées, cryptées puis compressées ou cryptées et compressées au même moment. Comme l'illustre la figure 1.16, il est plus avantageux d'effectuer la compression et le cryptage simultanément.

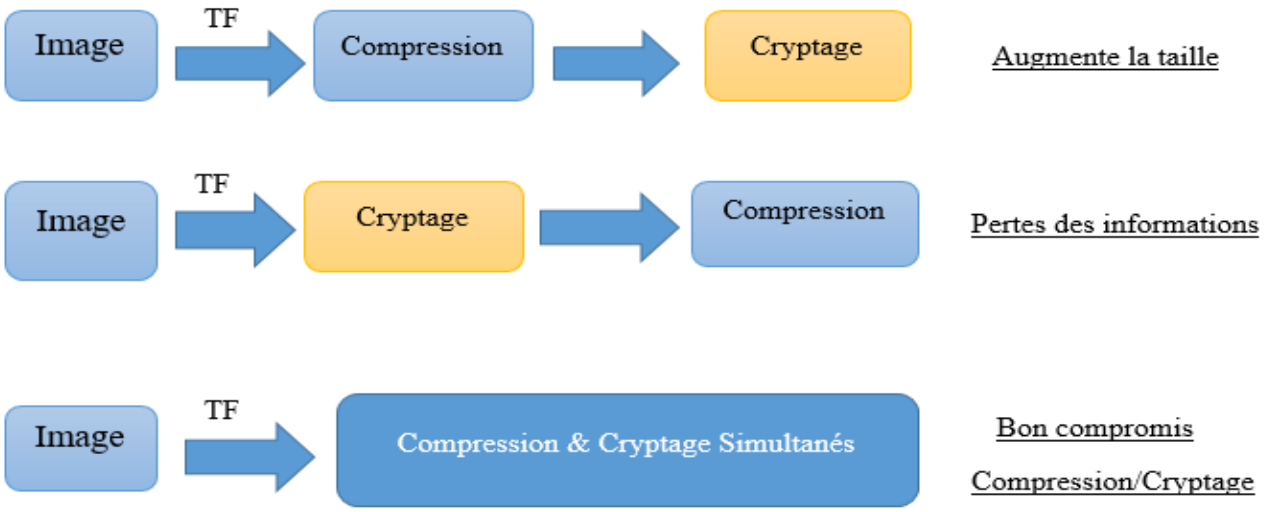


FIGURE 1.16 – Schématisation des techniques de compression et cryptage.

Dans la plupart des algorithmes basés sur la crypto-compression, la fusion d'images est réalisée au niveau de la phase compression, ce qui permet par la suite de réduire la quantité de données à transmettre. Nous présentons quelques exemples de ces algorithmes proposés dans la littérature.

Une nouvelle technique de crypto-compression a été développée par Ayman A. et al. [62]. Elle utilise le montage optique de la DCT (figure 1.17) et le principe du filtrage fréquentiel pour comprimer et crypter simultanément les images.

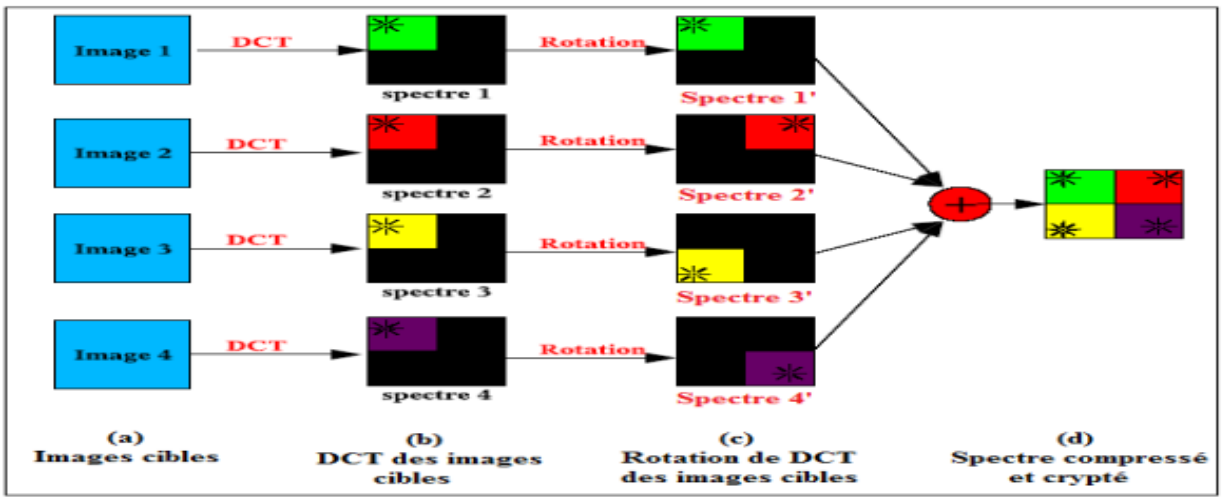


FIGURE 1.17 – Diagramme synoptique de la méthode de compression et cryptage simultanés [62].

Selon le schéma de la figure 1.17, la première étape du cryptage consiste à réaliser séparément les DCTs de différentes images cibles (images à comprimer et à crypter). En se basant sur la propriété de DCT, qui permet de regrouper les informations dans le coin haut gauche du spectre, un filtre passe bas est appliqué aux différents spectres. Après, il suffit de regrouper (par une simple opération d'addition par exemple) ces différents spectres filtrés pour n'en obtenir qu'un seul qui contient toutes les informations nécessaires pour reconstruire toutes les images cibles en sortie. Cependant, cette technique présente des limites avec l'apparition d'un problème de saturation, surtout lorsque l'on veut augmenter le nombre d'images à multiplexer ensemble. Cette saturation est due au fait que l'on regroupe les différents spectres dans la même zone (coin haut gauche) du plan spectral, d'où la nécessité d'optimiser ce regroupement. Pour réaliser cette optimisation, nous pouvons observer qu'après le filtrage des différents spectres, une petite partie du plan spectral est utilisée, laissant ainsi les autres parties libres. Dans ces parties libres, nous pouvons mettre les autres spectres, mais il faut auparavant réaliser une rotation à 90 degrés (figure 1.17-c) à chaque fois pour mieux optimiser le plan spectral et garder le maximum de pixels représentatifs des spectres.

Toujours dans la perspective d'optimiser le nombre d'images à fusionner afin d'éviter le problème de chevauchement, M. Aldossari [84]a proposé une approche de compression par fusion spectrale. Cette technique utilise un nouveau critère pour chercher les informations pertinentes dans le spectre de l'image. Ce critère est basé sur l'utilisation de la taille utile du spectre de l'image (en anglais Root-Mean-Square-Duration). Cette taille utile (RMS) du spectre représente les pixels qui contiennent les informations importantes de l'image. Ainsi, en ne gardant qu'une partie du spectre, nous pouvons multiplexer un certain nombre de spectres d'image cible et ainsi obtenir une utilisation optimale du produit espace-bande passante (SBWP) dans le domaine de Fourier. La figure 1.18 présente le synoptique de la technique de cryptage proposée.

Lorsque les critères de segmentation sont bien choisis et que l'on effectue un décalage de chaque spectre avant la fusion, le nombre d'images à multiplexer augmente et la qualité des images reconstruites est satisfaisante (avec un MSE égal à 1.5×10^{-3} pour 4 images multiplexées). Pour un nombre d'images à multiplexer supérieur à 8, le seul critère de segmentation basé sur le Root Mean Square duration n'est plus suffisant pour obtenir un bon compromis entre le rapport de compression, la robustesse du cryptosystème et la qualité des images reconstruites.

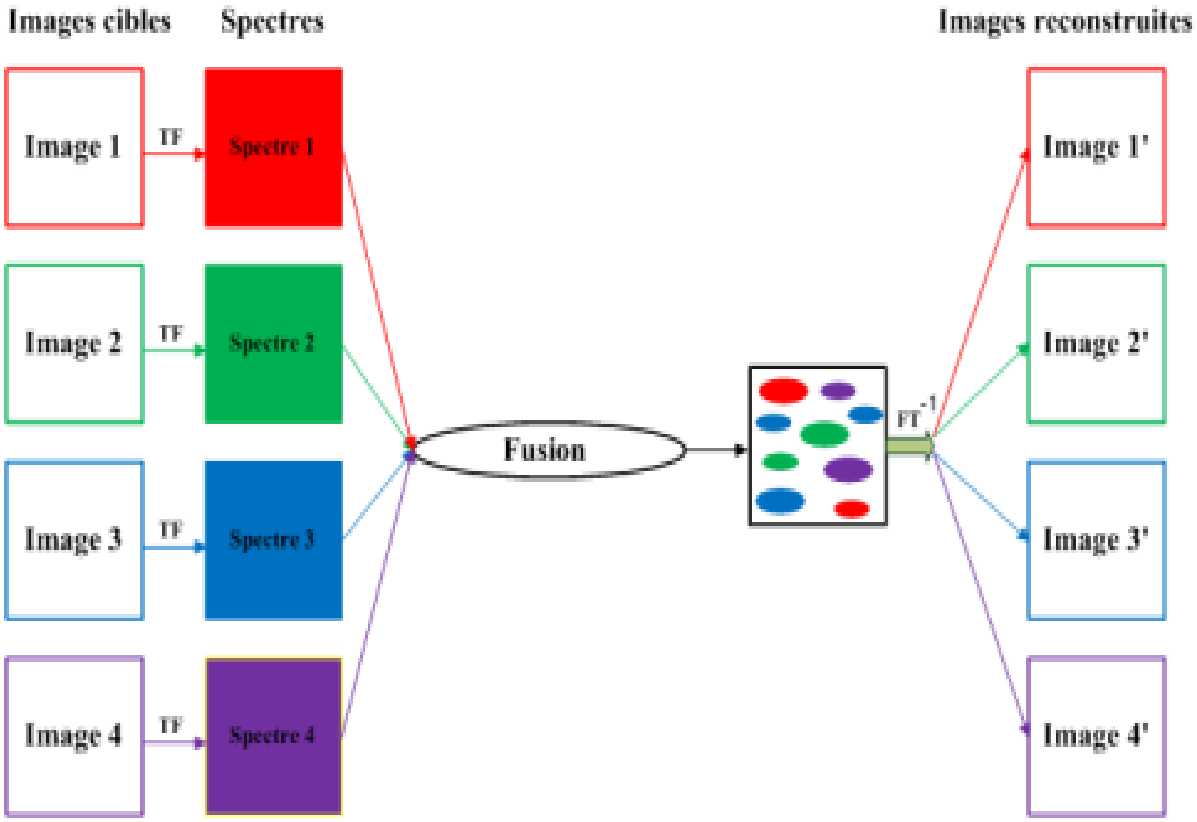


FIGURE 1.18 – Schéma synoptique de la méthode de fusion spectrale [84].

3. Méthodes construites selon le protocole permutation-diffusion.

De manière générale, les méthodes de fusion dans le domaine spectral présentent l'avantage de combiner plusieurs images en une en réduisant la quantité de données à transmettre, mais quelque fois, souffrent du problème de la qualité d'image reconstruite qui est dégradée ; mais aussi de la robustesse du cryptosystème. Pour celles basées sur la crypto-compression, un rapport de compression supérieur à 80 % entraîne une mauvaise qualité de l'image reconstruite. Dans le souci de renforcer la robustesse des algorithmes de chiffrement de grandes quantités d'images, des auteurs ont proposé des techniques de fusion ou mixage d'images dans le domaine spatial. La plupart de ces techniques sont construites suivant le schéma de permutation-diffusion.

Entre autres, Ayman et al. [62] ont proposé un nouvel algorithme qui mixe deux ou plusieurs images de différents types et tailles en utilisant une procédure de brouillage combinée

à la substitution par les boîtes S (en anglais S-box) pour effectuer un cryptage d'images sans perte. Ici, le processus de fusion des images combine le chiffrement par flux avec celui en bloc, au niveau du pixel. Les résultats expérimentaux obtenus ont révélé que le cryptosystème avait des performances acceptables, mais certaines améliorations doivent être effectuées pour renforcer la robustesse du cryptosystème. Également, les auteurs n'ont pas renseigné sur la valeur des métriques telles que l'entropie, l'UACI et le NPCR qui sont essentielles pour apprécier le niveau de sécurité de l'algorithme.

Gui-Liang et al. [58] ont élaboré un algorithme de cryptage d'éléments d'images mixés basé sur une technique de courbe elliptique. Cet algorithme a introduit le concept d'élément d'image composite (CIE). Les résultats expérimentaux et l'analyse théorique montrent que l'algorithme présente un large espace clé et peut produire un niveau de sécurité élevé en ce qui concerne l'interaction des informations sur un réseau de communication, mais le temps de cryptage et de décryptage est considérable.

Récemment, Maher J. et Ayman [54] ont proposé un algorithme pour améliorer un système de fusion, compression et de cryptage optique simultanés (SFCE) existant (Alfalou et al., 2011) en terme de temps de calcul, d'occupation de la bande passante et de robustesse du chiffrement. Les auteurs dans leur approche de cryptage ont utilisé une forme approximative de transformation discrète en cosinus (DCT) au niveau de la fusion afin de diminuer le temps de calcul. Par la suite, les cartes de Henon et Skew-Tent ont été utilisées pour réaliser la phase de confusion (le long des lignes et des colonnes) et les effets de diffusion. La méthode proposée est robuste contre les principales attaques et permet la transmission simultanée de plusieurs images de même taille.

Xiaoqiang Z. [56], [57] a proposé deux techniques de cryptage d'images : l'une basée sur le mixage d'éléments d'images et la permutation ; l'autre basée sur le mixage d'éléments d'images par le chaos. Ces deux méthodes introduisent le concept d'éléments d'images pures et d'images mixtes, et ont l'avantage de chiffrer plusieurs types d'images simultanément, avec une grande efficacité et en un temps réduit. Dans le processus de cryptage, les images cibles sont combinées en une grande image qui est découpée en blocs de petites tailles, puis l'image cryptée est obtenue après mélange des sous blocs de différentes images cibles, suivie d'une étape de diffusion. Le schéma de la figure 1.19 détaille les étapes du processus de chiffrement. En analysant les méthodes proposées, l'on constate que l'espace de clés n'est pas assez large ; de plus, il se pose un problème de taille de l'image à transmettre qui devient de plus en plus grande lorsque le nombre d'images cibles augmente.

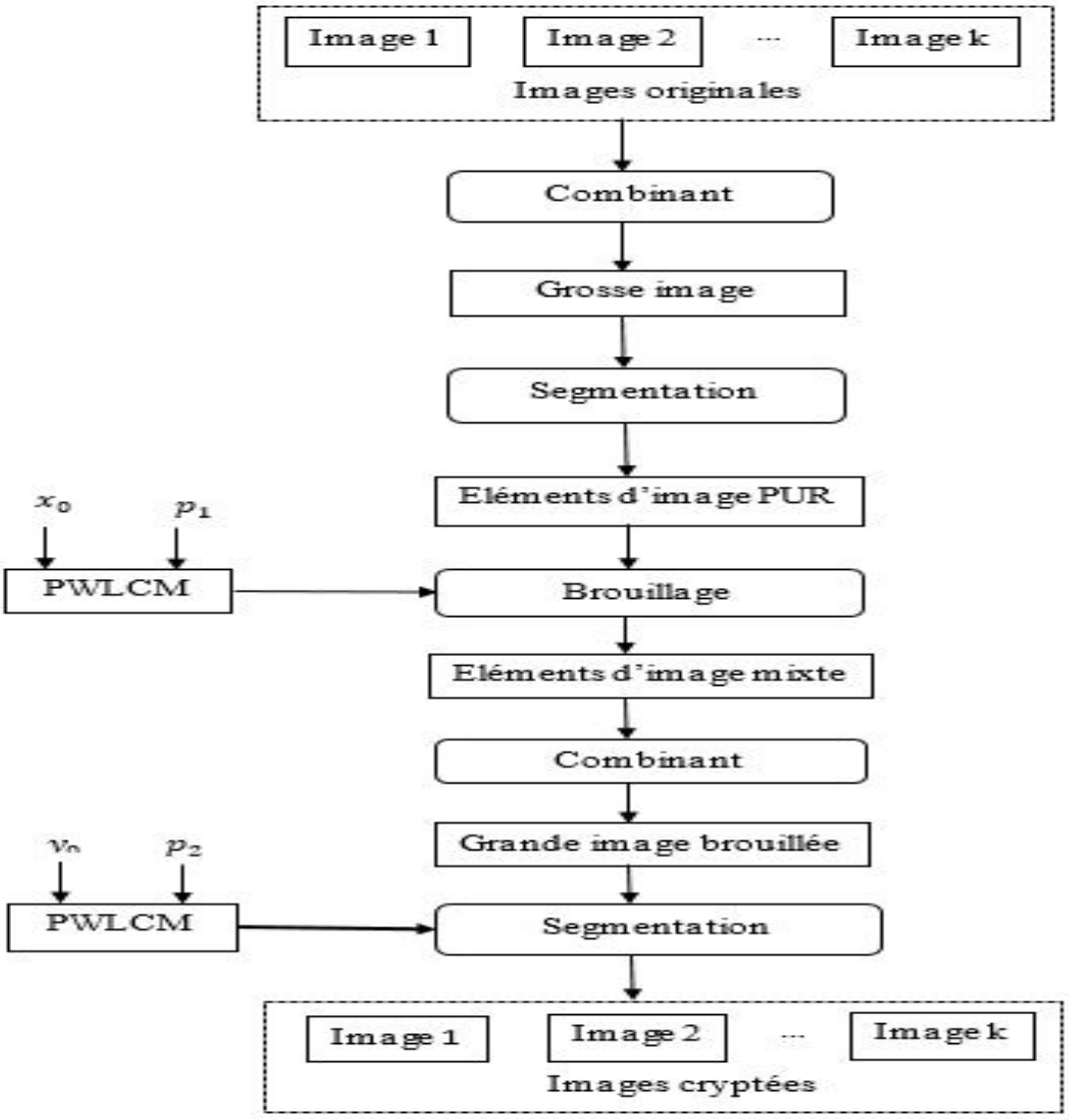


FIGURE 1.19 – Schéma de chiffrement basé sur le mixage d'éléments d'images et le chaos [57].

Fort de ce qui précède, il est possible de construire des cryptosystèmes robustes et efficients basés sur la fusion ou le mixage d'images qui permettent un bon compromis entre le nombre d'images à transmettre, la robustesse du système et la qualité des images reconstruites. Cet enjeu fera l'objet d'un développement au chapitre 2.

1.5.2.3 Cryptanalyse

Avec l'évolution des nouvelles technologies, plusieurs algorithmes dédiés au cryptage de données (texte, images, vidéos) ont été mis en œuvre par des chercheurs, sauf que ces algorithmes ne résistent pas tous aux différentes attaques des intrus. Selon le principe de Kerckhoffs

(1883), la robustesse ou le niveau de sécurité d'un algorithme ne devrait pas résider dans la technique de chiffrement/déchiffrement utilisée, mais au niveau de la longueur de la clé. Plusieurs attaques sont souvent utilisées par les inconnus pour éprouver le niveau de sécurité des algorithmes, dans le but découvrir une partie ou la totalité de l'information cryptée. Ce procédé est appelé cryptanalyse, technique qui consiste à déduire un texte en clair d'un texte chiffré sans posséder la clé de chiffrement. Les différentes attaques régulièrement utilisées sont adaptées aux techniques de chiffrement et peuvent être groupées comme suit :

- **L'attaque à texte chiffré seul (Cipher text-only attack)** : L'attaquant a connaissance du texte chiffré de plusieurs messages.
- **L'attaque à texte clair connu (Known-plain textattack)** : Le cryptanalyste a accès à plusieurs textes chiffrés ainsi qu'aux textes clairs correspondants.
- **L'attaque à texte clair choisi (Chosen-plain text attack** : L'attaquant a accès à l'algorithme de chiffrement. Il l'utilise pour générer des couples (Xi, Yi) de son choix. La différence principale par rapport l'attaque à texte clair connu est que le cryptanalyste peut choisir le texte à chiffrer.
- **L'attaque à texte chiffré choisi (Adaptative-plain text attack** : Le cryptanalyste a accès à l'algorithme de décryptage. Il peut choisir les textes à déchiffrer sans connaître la clef.
- **L'attaque par force brute (Brute-force attack** : ou l'attaque exhaustive : l'attaquant essaie toutes combinaisons de clefs possibles jusqu'à l'obtention du texte clair.
- **L'attaque par canaux auxiliaires** : Toutes les façons d'analyser les propriétés inattendues d'un algorithme sont prises en compte pour réussir à casser le cryptosystème. Dans les algorithmes de chiffrement implémentés en hardware, par exemple, la consommation électrique pour chaque type de calcul du chiffrement peut être utile pour déduire certaines informations de la clef.
- **L'attaque algébrique** : Les attaques algébriques sont des attaques à texte clair connu qui exploitent des relations algébriques entre les bits du texte clair, ceux du chiffré et ceux de la clef secrète. La connaissance de plusieurs couples clairs-chiffrés fournit donc un système d'équations dont les inconnues sont les bits de la clef secrète. Ces derniers peuvent alors être retrouvés en résolvant le système, ce qui est possible si le système est de degré faible, de petite taille ou qu'il possède une structure particulière.

Le développement de la cryptanalyse a entrainé ces dernières années la création de nouvelles techniques modernes de chiffrement. A cet effet, les développeurs de cryptosystèmes essayent de classer les algorithmes de cryptage en fonction du niveau de sécurité face aux attaquants. Sous cet angle, nous pouvons grouper les techniques de cryptanalyse selon deux familles : la cryptanalyse différentielle et celle linéaire.

- **Cryptanalyse différentielle**

Elle a été mise au point par Eli Biham et Adi Shamir en 1991. Elle permet de trouver la clef en utilisant une quantité de textes clairs. L'idée est de fournir comme entrée des textes clairs avec de légères différences (un bit par exemple). Ensuite, on analyse statistiquement le comportement des sorties selon les entrées pour retrouver la clef. En regardant comment les différences en entrée affectent les sorties, on peut établir des règles statistiques. Il existe plusieurs variantes de cryptanalyse différentielle. Nous distinguons : la différentielle tronquée, différentielle d'ordre supérieur et la différentielle impossible [85].

- **Cryptanalyse linéaire**

Elle a été inventée par Mitsuru M. en 1993 [85]. Elle nécessite une quantité N de couples (texte clair, texte chiffré), tous chiffrés avec la même clef. Le principe est que le même message soit chiffré plusieurs fois avec des clefs différentes pour construire une immense table (téraoctet) qui contient toutes les versions chiffrées de ce message. Lors d'une interception d'un message chiffré, on peut le retrouver dans la table et obtenir la clef qui avait été utilisée pour le cryptage. Généralement, cette attaque est difficile d'être menée.

Conclusion

Tout au long de ce chapitre, nous avons fourni les informations générales sur la cryptographie, notamment son origine et évolution, ses concepts de base et outils. Nous avons aussi présenté les techniques de chiffrement à l'ère moderne et avons mis une emphase sur les crypto-systèmes basés sur la fusion ou le mixage d'images. Ces derniers présentaient l'intérêt d'effectuer le cryptage de plusieurs images simultanément, tant dans le domaine spatial que fréquentiel en recherchant un bon compromis entre robustesse, qualité des images reconstruites et quantité d'images transmises. Dans le prochain chapitre, nous ferons une présentation des algorithmes que nous avons proposé, basés sur le mixage d'images, puis décrirons les métriques d'évaluation d'un bon crypto-système.

Chapitre II

Etude méthodologique du cryptage par fusion/mixage d'images et élaboration de deux algorithmes basés sur la fusion

Introduction

Avec l'échange de grandes quantités d'images via les différents canaux de communication peu sécurisés ainsi que le développement considérable de la cryptanalyse, de nouveaux crypto-systèmes basés sur la fusion/mixage d'images ont été développés. Ces derniers présentent un intérêt de pouvoir combiner plusieurs images en les mélangeant en une seule avant la transmission, ce qui rend complexe la reconstitution des images cibles sans la connaissance des clés. Plusieurs algorithmes ont été mis en œuvre, tant dans le domaine spatial que spectral, mais l'on observe dans la plupart des cas que les résultats de chiffrement obtenus ne donnent pas un bon compromis entre quantité d'images transmises, qualité d'images reconstruites, robustesse et temps de calcul. En guise de contribution, nous proposons dans ce chapitre deux algorithmes de cryptage basés sur la fusion d'images : le premier effectue la fusion de deux images chiffrées séparément pour obtenir des images hybrides. Le second quant à lui réalise deux niveaux de fusion d'images cibles en exploitant la DCT, la permutation au niveau des blocs et la diffusion basée sur une relation non linéaire calquée sur le système de Kramer. Les cartes de données sont utilisées dans les deux algorithmes proposés comme générateurs de nombres pseudo-aléatoires (GNSA), utiles dans les processus de génération de clés et des étapes de chiffrement. Le présent chapitre est organisé comme suit : nous présenterons premièrement l'organisation des crypto-systèmes basés sur la fusion d'images et les outils physiques et mathématiques utilisés pour la construction de notre crypto-système. Ensuite, nous ferons la description générale des deux algorithmes proposés. Le chapitre se referme par la présentation des outils d'analyse de sécurité d'un bon crypto-système.

2.1 Prototype de crypto-systèmes basés sur la fusion d'images

Les techniques de fusion ou mixage d'images pour les applications de chiffrement d'images ne sont pas toutes bâties selon le même protocole, mais reposent sur trois points principaux : le niveau de fusion, le domaine de fusion et la technique de fusion utilisée.

2.1.1 Le niveau de fusion

Comme évoqué au chapitre un, la fusion d'images peut se faire au niveau des caractéristiques (couleur, résolution, etc.), du pixel et de la décision (selon certains critères particuliers). En cryptographie, la fusion d'images au niveau du pixel est plus pratique, car cette dernière rend possible la reconstitution des données initiales sans pertes d'informations.

2.1.2 Le domaine de fusion

En parcourant la littérature, la fusion d'images en cryptographie se fait soit dans le domaine spectral (des fréquences), soit dans le domaine spatial. Le choix de chacun des domaines présente des atouts tout comme des limitations.

- Dans le domaine fréquentiel, les images cibles à chiffrer sont d'abord transformées dans le domaine fréquentiel, puis combinées en une image qui contient le spectre de toutes les images. Les transformations généralement utilisées pour la fusion spectrale sont : la transformation de Fourier, la Transformation de Fourier Fractionnaire, la Transformation de Fourier Discrète, la Transformée Discrète en Cosinus, la transformation en Ondelettes, etc. Dans le domaine fréquentiel, il est plus aisé de fusionner un grand nombre d'images en un temps réduit, mais cela entraine les pertes de données lors du processus de décryptage (la qualité d'images reconstruites n'est pas toujours bonne).
- Dans le domaine spatial, les images originales sont combinées en une image par un processus de multiplexage ou un mode de fusion itératif. Un atout majeur est que les images décryptées ne connaissent pas de pertes d'informations. Toutefois, si le nombre d'images à chiffrer devient important, le temps de chiffrage sera conséquent, car la taille de l'image mixée à chiffrer devient très grande.

2.1.3 La technique de fusion utilisée

Spécifiquement pour le cryptage d'images, la technique de fusion utilisée devrait être réversible afin de rendre possible le processus de décryptage, et ne devrait pas être source de pertes de données. Puisque la robustesse de l'algorithme en cryptographie repose sur la longueur de la clé, la méthode choisie, même si elle est réversible devrait avoir des paramètres servant de clés au crypto-système. Parmi les techniques classiques de fusion d'images, peu d'entre elles sont

réversibles (DCT, DWT, Réseaux artificiels de Neurones, etc.). Pour utiliser ces dernières afin de construire des algorithmes de chiffrage robustes, plusieurs auteurs les associent avec d'autres algorithmes de cryptage. Les algorithmes de cryptage d'images développés ces dernières années utilisent des cartes de données comme générateurs de nombres pseudo-aléatoires, ceci compte tenu de leurs avantages.
Dans le prochain paragraphe, nous décrirons quelques travaux importants et récents, effectuant le cryptage d'mages par fusion, tant dans le domaine spatial que fréquentiel. Cela permettra de mettre en valeur les différents outils utilisés dans leurs procédés de fusion.

2.2 Quelques travaux récents sur le cryptage par fusion/mixage d'images

2.2.1 Algorithme de chiffrement de A. A. Karawia

Karawia a proposé un algorithme de cryptage de plusieurs images en utilisant le concept d'éléments d'images mixés et la carte chaotique économique (CEM) de dimension deux (voir équation 2.1) [86]. Les images d'entrée sont premièrement combinées en une grande image qui est subdivisée en sous blocks de petite taille. Ces derniers sont ensuite permutés à l'aide de la CEM, puis l'image mixée obtenue est brouillée par la carte logistique. L'image résultante est subdivisée selon les différentes tailles d'images originales.

$$\begin{cases} \alpha_{n+1} = \alpha_n + k\left[a - c - \frac{b\alpha_n}{\gamma_n} - b\log(\gamma_n)\right] \\ \beta_{n+1} = \beta_n + k\left[a - c - \frac{b\beta_n}{\gamma_n} - b\log(\gamma_n)\right] \end{cases} \tag{2.1}$$

$\gamma_n = \alpha_n + \beta_n, \quad n = 0, 1, 2, ...$
En terme de performance, l'algorithme proposé, inspiré de celui de Xiaoqiang Z. et Xuesong W. [57] présente un espace de clés large (10^{210}), et présente de bonnes valeurs pour les métriques UACI, NPCR, entropie, etc. Bien que la structure de l'algorithme soit simple et permet de chiffrer plusieurs images en un temps réduit, le choix des cartes de données utilisées pose un problème. En fait, les cartes logistique et CEM ont une plage de paramètre de contrôle réduite, une distribution non uniforme des fréquences de nombres générés et une faible ergodicité (voir figure 2.1). Etant donné que l'essentiel des paramètres de la clé du crypto-système proposé repose sur les conditions initiales et paramètre de contrôle des cartes de données pas assez fiables, les attaques de cryptanalyse à image claire choisie et image chiffrée choisie peuvent bien connaître du succès sur cet algorithme. Précisément, l'auteur n'a pas mené les tests reposant sur ces deux attaques de cryptanalyse mentionnées précédemment.

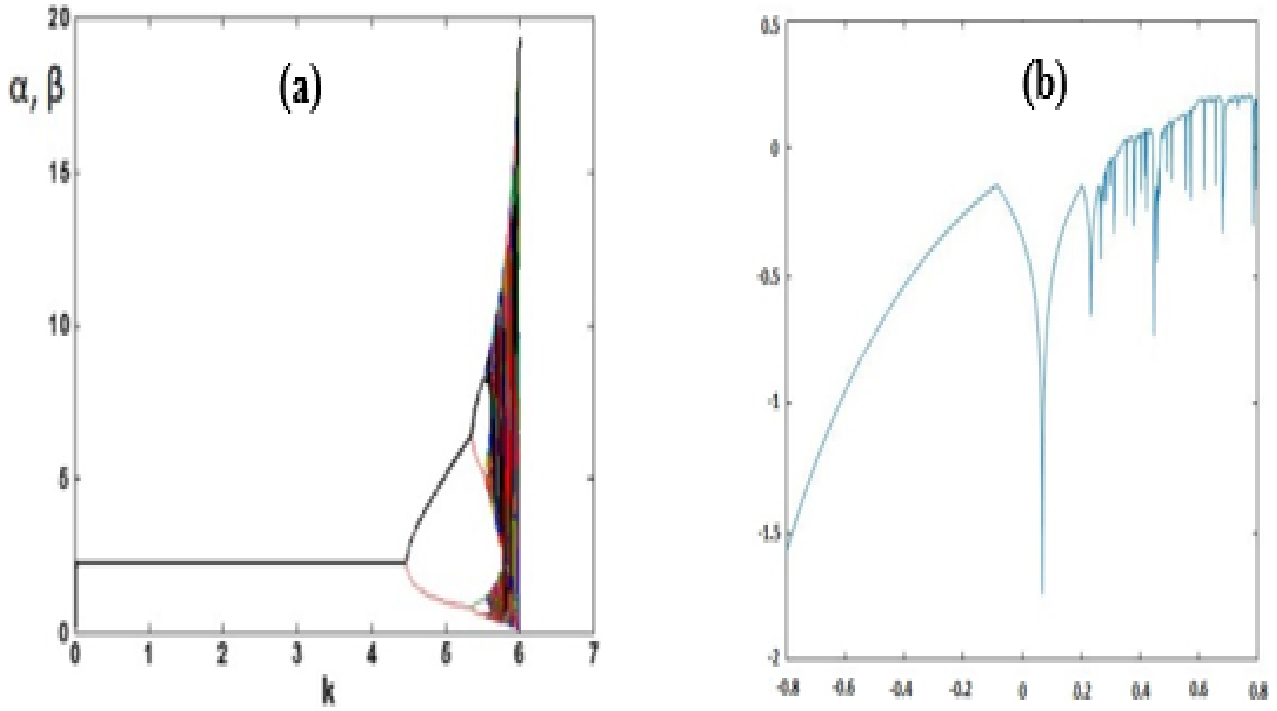


FIGURE 2.1 – Diagramme de bifurcation et exposant de Lyapunov de la Carte Chaotique Economique (CEM). (a) diagramme de bifurcation pour a = 3, b = 1, c = 1, $a0 = 0.19$, $b0 = 0.15$ et $k \in [0, 6.0001]$, (b) Exposant de Lyapunov [86].

2.2.2 Algorithme de chiffrement de Maher J. et Ayman A.

Maher Jridi et Ayman Alfalou ont proposé une technique de fusion, compression et cryptage simultanée pour améliorer celle proposée par Alfalou et al. [87] en termes de temps de cryptage, de bande passante et robustesse. Dans l'algorithme précédent, les images sont d'abord transformées dans le domaine spectral par la DCT, puis chacune d'elle est multipliée par un filtre approprié avant la phase de fusion. L'image fusionnée est combinée avec un masque de phase afin de donner une image compressée et cryptée (voir figure 2.2) [54]. L'algorithme précédent étant vulnérable aux attaques à force brute, sa version améliorée par les mêmes auteurs consiste à conserver l'image fusionnée du premier algorithme, puis à base des cartes de Henon et Skew-Tent effectuer la permutation des pixels suivant les lignes et colonnes et renforcer avec la phase de diffusion. Le schéma simplifié de l'algorithme de chiffrement est illustré à la figure 2.3. Le choix des cartes utilisées par les auteurs a conduit aux bonnes performances du crypto-système proposé qui est robuste et rapide. De plus, la taille de l'image fusionnée est presque égale à celle d'une image originale, d'où la quantité de données transmises est considérablement réduite.

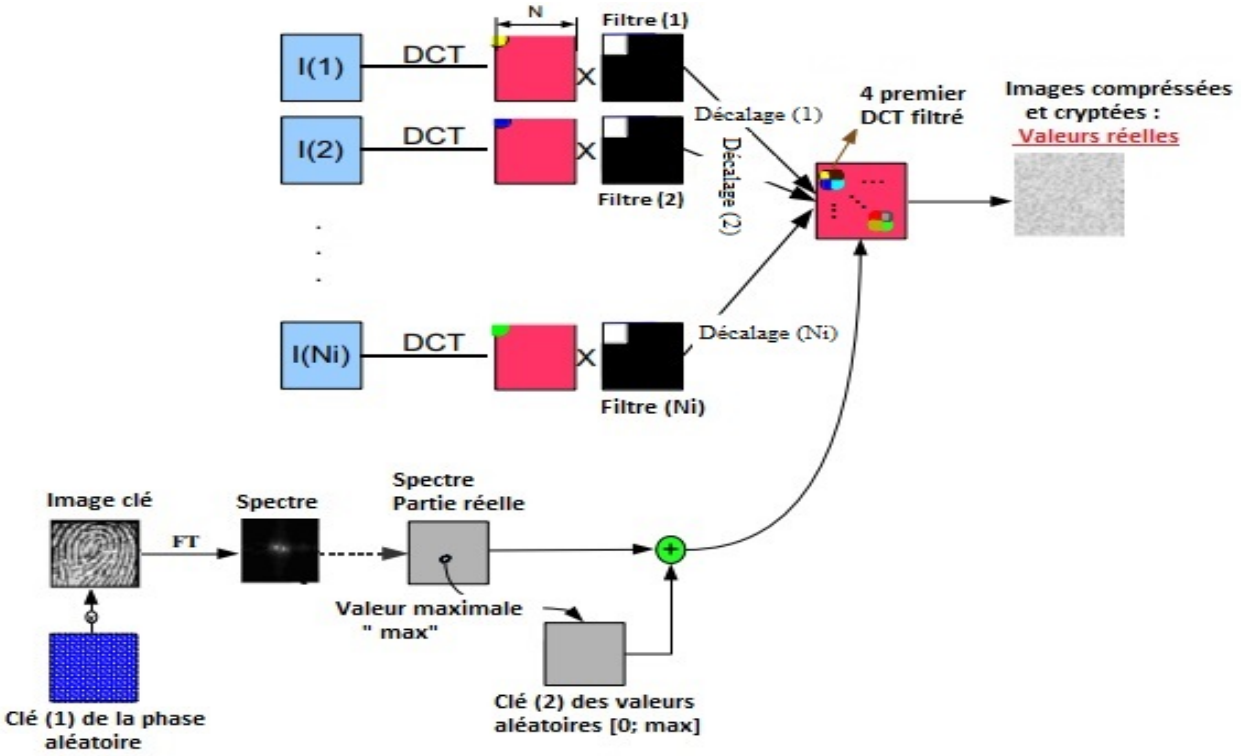


FIGURE 2.2 – Schéma de cryptage proposé par Alfalou et al. [55].

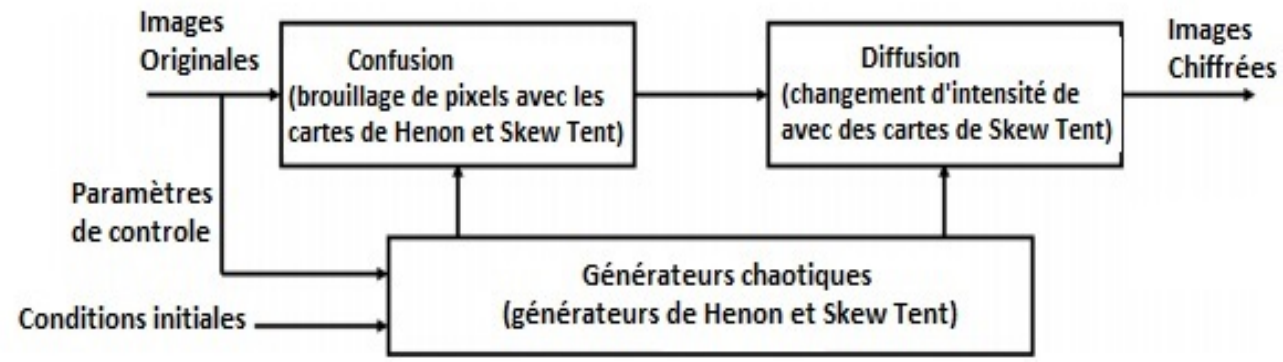


FIGURE 2.3 – Schéma de cryptage amélioré proposé par Maher J. et Ayman A. [87]

2.2.3 Algorithme de chiffrement de Xiaoqiang Z. et Xuesong W.

Xiaoqiang Zang et Xuesong Wang ont proposé un algorithme de cryptage d'images utilisant les courbes elliptiques (EEC) [88]. Le crypto-système proposé est asymétrique, et le protocole d'échange de clés entre l'émetteur et le récepteur est basé sur la technique de clé publique de

Diffie–Hellman. L'ensemble d'images à chiffrer est combinée en un ensemble d'images mixées (figure 2.4-a) avant suivre le processus de chiffrement tel que présenté à la figure 2.4-b. L'algorithme présenté est robuste et sensible à la clé et à l'image en clair. Toutefois, le temps de chiffrement doit être optimisé pour que l'algorithme soit adapté au cryptage d'images en temps réel.

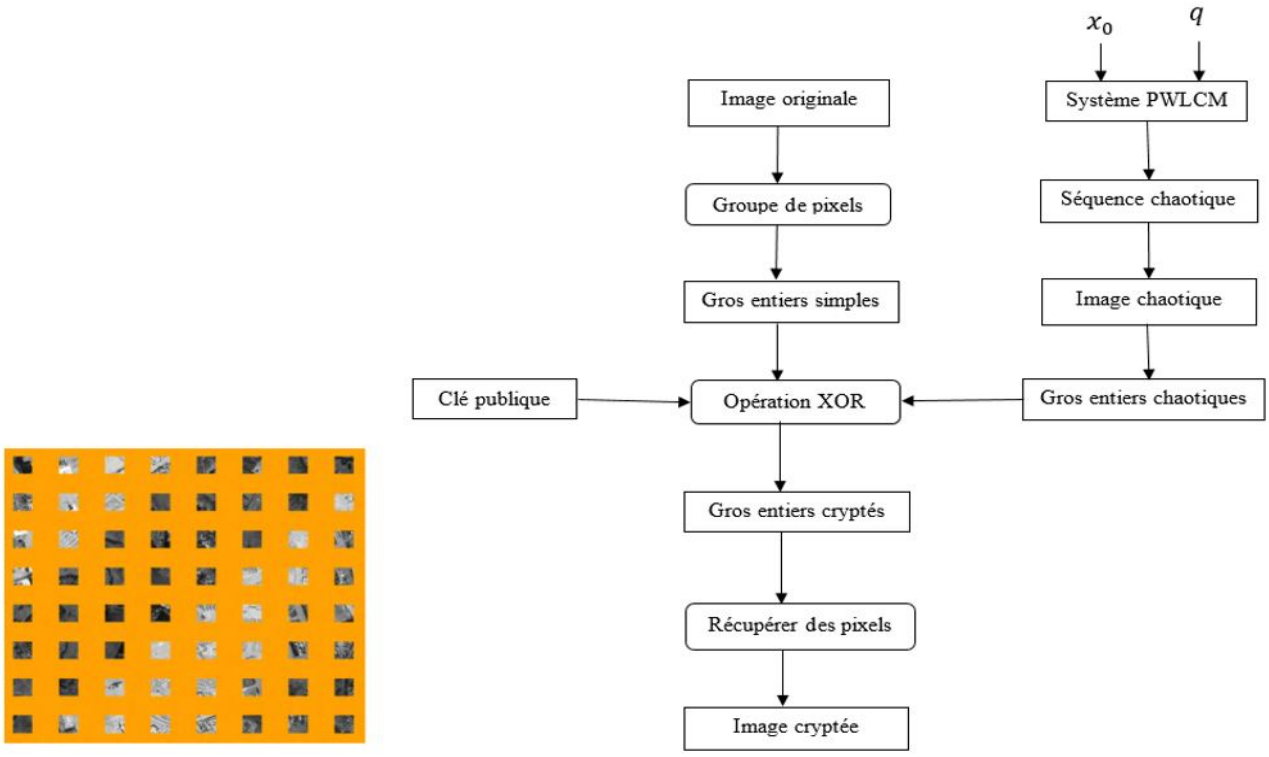


FIGURE 2.4 – Image d'entrée et algorithme de cryptage. (a) image en texte clair (ensemble d'images à chiffrer), (b) Algorithme de chiffrement [88].

2.2.4 Algorithme de chiffrement de Yi Qin et al.

Yi Qin et coauteurs ont proposé un crypto-système optique basé sur la théorie de diffraction utilisant la fusion spectrale d'images et des opérations non-linéaires. Dans le processus de cryptage, les spectres d'images obtenus par la DCT sont extraits, puis combinés en une image qui sera multipliée par une matrice issue d'un ensemble d'opérations non linéaires. L'image obtenue est aussi multipliée par un masque de phase pour obtenir l'image chiffrée [50]. Les figures 2.5 et 2.6 présentent respectivement le schéma de cryptage et un exemple d'images chiffrées avec cet algorithme. Après analyse de cet algorithme, l'on relève qu'il permet de chiffrer plusieurs images par un procédé de fusion en un temps réduit, mais l'espace de clés

n'est pas grand et la qualité d'images reconstruites se dégrade considérablement pour plus de quatre images à chiffrer.

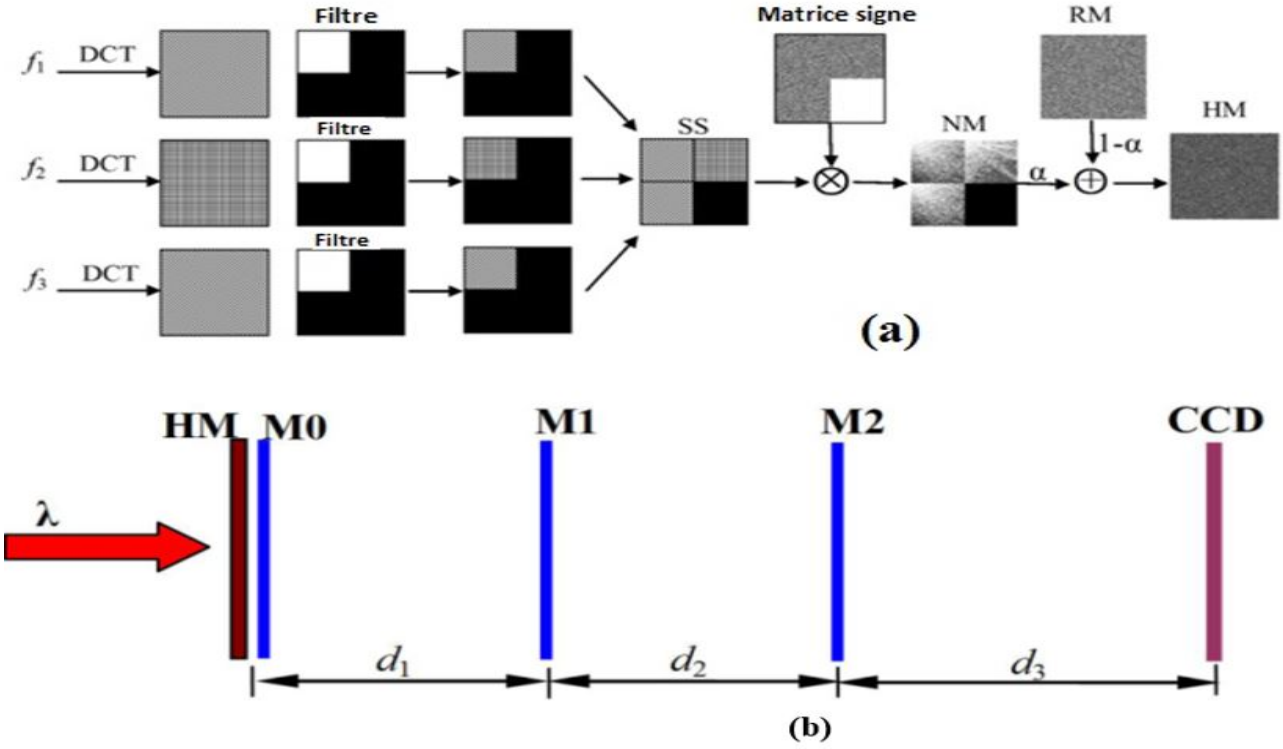


FIGURE 2.5 – Schéma de cryptage de l'algorithme. (a) premier niveau de cryptage, (b) deuxième niveau de cryptage (dispositif optique) où M0, M1et M2 sont des masques de phase et CCD l'écran où se forme l'image chiffrée [50].

Légende : f_1, f_2 et f_3 sont les trois images à chiffrer de taille $N \times N$; (SS) représente le spectre des images fusionnées ; les valeurs de (SS) sont constituées de valeurs positives, négatives et nulles (en noir). (SS) est transformée en une matrice non négative (NM) par la relation

$$NM(u,v) = SS(u,v)s(u,v) \tag{2.2}$$

Avec $s(u,v)$ est une matrice signée définie par :

$$s(u,v) = \begin{cases} 1 & SS(u,v) \geq 0 \\ -1 & SS(u,v) < 0 \end{cases} \tag{2.3}$$

$s(u, v)$ étant une fonction à sens unique. La matrice NM est transformée en une autre matrice HM par la relation 2.4.

$$HM(u, v) = \alpha NM(u, v) + (1 - \alpha)RM(u, v) \tag{2.4}$$

Où $RM(u, v)$ est une matrice à valeurs réelles et aléatoires comprises entre [0,1], α est un coefficient constant compris entre 0 et 1 qui indique la taille de chacune des images du côté droit de l'équation (2.4).

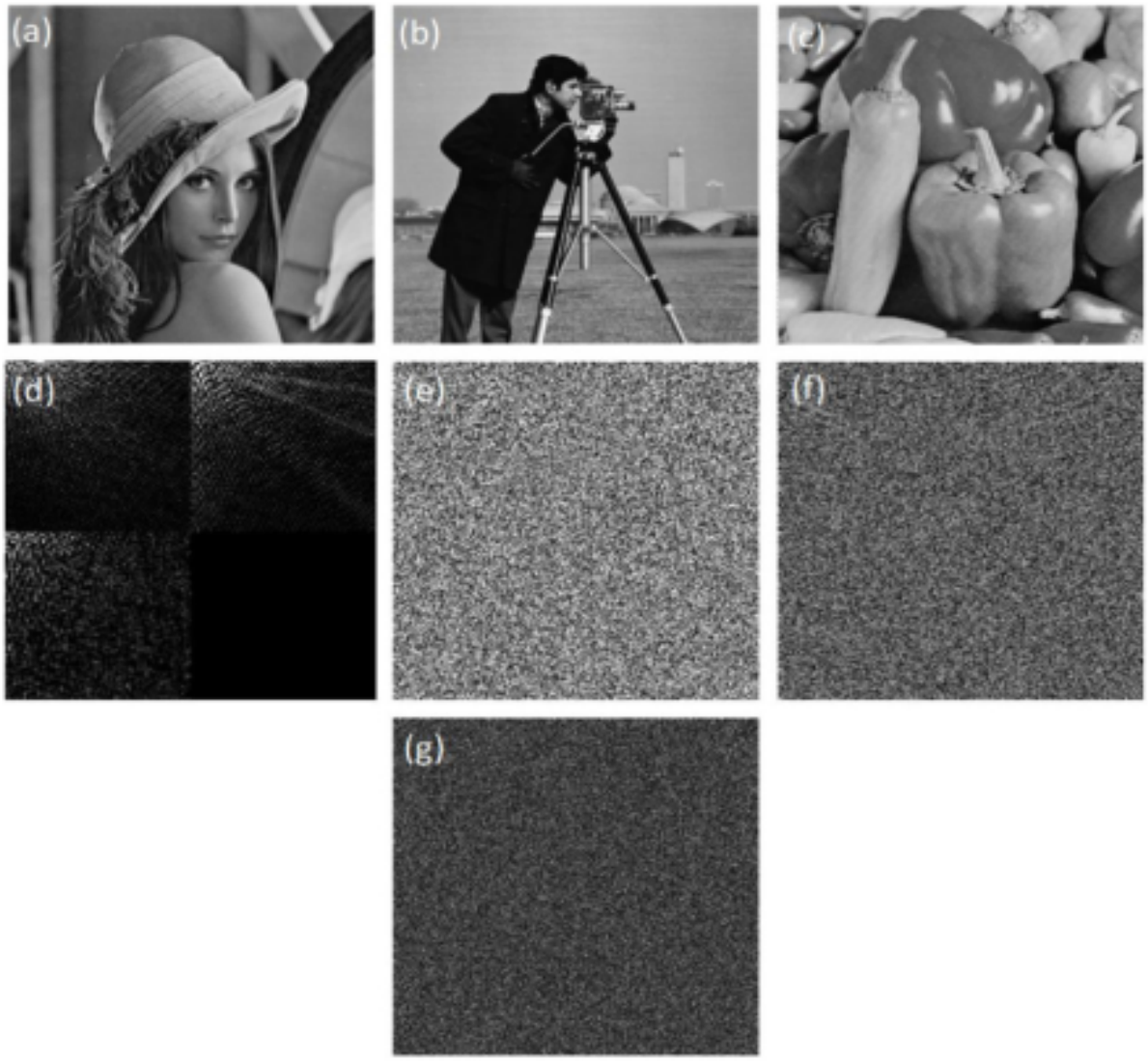


FIGURE 2.6 – Images chiffrées avec l'algorithme de Yi Q. et al. [50]. (a, b, c) Images originales, (d) spectre d'images fusionnées, (e) Image Lena cryptée, (f) Image Cameraman cryptée, (g) Image Pepper cryptée.

2.3 Synthèse des travaux de cryptage portant sur la fusion d'images et nouvelles contributions.

Après analyse des algorithmes de cryptage d'images proposés ces dernières années, il ressort que le choix de la technique de chiffrement est déterminant pour la robustesse du système, le nombre d'images à chiffrer ainsi que la qualité d'images décryptées. Plusieurs outils sont utilisés dans ces algorithmes, notamment la DCT, la Transformée de Fourier Fractionnaire, la notion de filtrage fréquentiel, la théorie sur les opérations matricielles, les cartes de données utilisées comme générateurs de nombres pseudo-aléatoires. Pour les algorithmes effectuant la fusion d'images dans le domaine spectral, elles peuvent chiffrer plusieurs images en un temps réduit, mais pour la plupart présente des limites au niveau de la robustesse et de la qualité d'images reconstruites. Quant aux algorithmes effectuant la fusion dans le domaine spatial, ils présentent une bonne robustesse et conservent une bonne qualité d'images après le décryptage. En revanche, ces derniers prennent un temps considérable pour chiffrer une grande quantité d'images. En conséquence, en adoptant une structure de chiffrement qui effectue la fusion d'images dans le domaine spectral puis spatial, cela permettrait d'avoir comme résultat un crypto-système robuste, rapide et conservant une bonne qualité d'images décryptées. C'est dans l'optique d'apporter des éléments de solution aux problèmes précédemment évoqués que nous proposons dans la suite deux crypto-systèmes : le premier effectue la fusion d'images par une approche hybride dans le domaine spatial et le second associe la fusion dans les domaines fréquentiel et spatial. Nous commençons par présenter les outils qui ont servi à la construction de nos algorithmes : les cartes de données, la notion de filtrage fréquentiel, la DCT, la théorie sur les opérations matricielles, etc.

2.4 Outils utilisés pour bâtir les deux crypto-systèmes proposés

2.4.1 Les cartes de données

Les cartes de données sont des systèmes le plus souvent utilisés pour la génération de séquences de nombres pseudo-aléatoires, nécessaires dans le processus de chiffrement des données. Leur extrême sensibilité aux petites variations des conditions initiales et paramètre de contrôle est un atout pour la génération de clés. Les cartes données chaotiques sont très utilisées peuvent être analogiques (exemple, les oscillateurs Duffing, Colpits, Hartley, Schua, Lorentz) ou discrètes ; mais dans ces travaux, nous utiliserons les cartes discrètes pour leurs nombreux avantages. Entre autres, ces dernières sont adaptées pour une implémentation software, régies par des relations mathématiques et beaucoup plus aisées à contrôler que les systèmes

analogiques, car ces derniers sont souvent soumis au bruit qui limite leur espace de clé.

2.4.1.1 Carte de Henon

La carte de Henon est un système dynamique à temps discret qui présente un comportement chaotique et qui est défini par la relation 2.5 suivante :

$$\begin{cases} x_{i+1} = 1 - ax_i^2 + y_i \\ y_{i+1} = bx_i \end{cases} \tag{2.5}$$

Où a et b sont les paramètres de bifurcation, $x0$ et $y0$ sont les conditions initiales. Dans la pratique, on considère souvent $x_0 = y_0 = 0$. Pour observer le comportement chaotique du système, nous choisissons a=1.4 et $b = 0.3$. Afin d'obtenir le diagramme de bifurcation, on maintient le paramètre $b = 0.3$ et l'on fait varier le paramètre a de 0 à $1,4$. L'attracteur et le diagramme de bifurcation de Henon sont illustrés à la figure 2.7. Il peut être observé que la première bifurcation se produit autour de $a = 0.362$ suivie d'une double bifurcation à $a = 0.91$. De $a = 0.91$ à 1.08, des doubles bifurcations successives se reproduisent périodiquement. Au-delà d'une valeur de $a > 1.1$, la périodicité change en comportement chaotique.

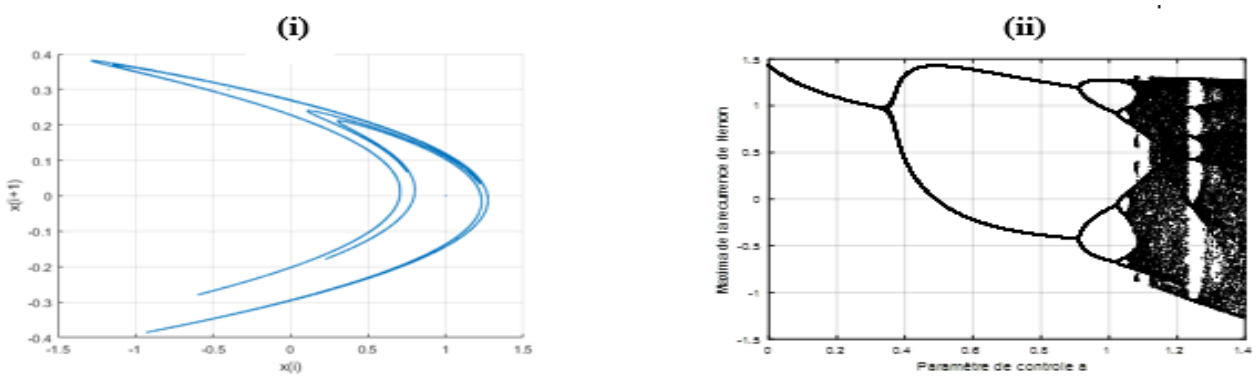


FIGURE 2.7 – Carte de Henon :(i) -Attracteur de Henon pour $a = 1.4$, $b = 0.3$; (ii)- Diagramme de bifurcation pour $b = 0.3$

2.4.1.2 Carte Logistique-May

Les cartes logistiques et May sont deux systèmes chaotiques beaucoup utilisés dans les crypto-systèmes pour leur simplicité et leur extrême sensibilité aux petites variations des conditions initiales. Elles sont respectivement définies par les relations 2.6 et 2.7 :
1) Carte logistique

$$x_{n+1} = rx_n(1 - x_n) \tag{2.6}$$

Où $x_n \in [0, 1]$ représente l'état discret de sortie du système, r est le paramètre de contrôle dont les valeurs appartiennent à l'intervalle [0, 4].

2) Carte May

$$x_{n+1} = x_n \exp(a(1 - x_n)) \tag{2.7}$$

Où $x_n \in [0, 10.9]$, a est le paramètre [0, 5].

3) Carte sine

$$x_{n+1} = b \sin(\pi x_n)/4 \tag{2.8}$$

Où $x_n \in [0, 4]$ et $b \in [0, 4]$.

La figure 2.8 illustre le diagramme de bifurcation ainsi que l'exposant de lyapunov des différentes cartes Logistique, May et Sine. L'on peut y observer que distribution des valeurs de x est dense seulement sur une plage du paramètre de contrôle très réduite, ce qui met en exergue les zones périodiques sur le diagramme de bifurcation de ces cartes.

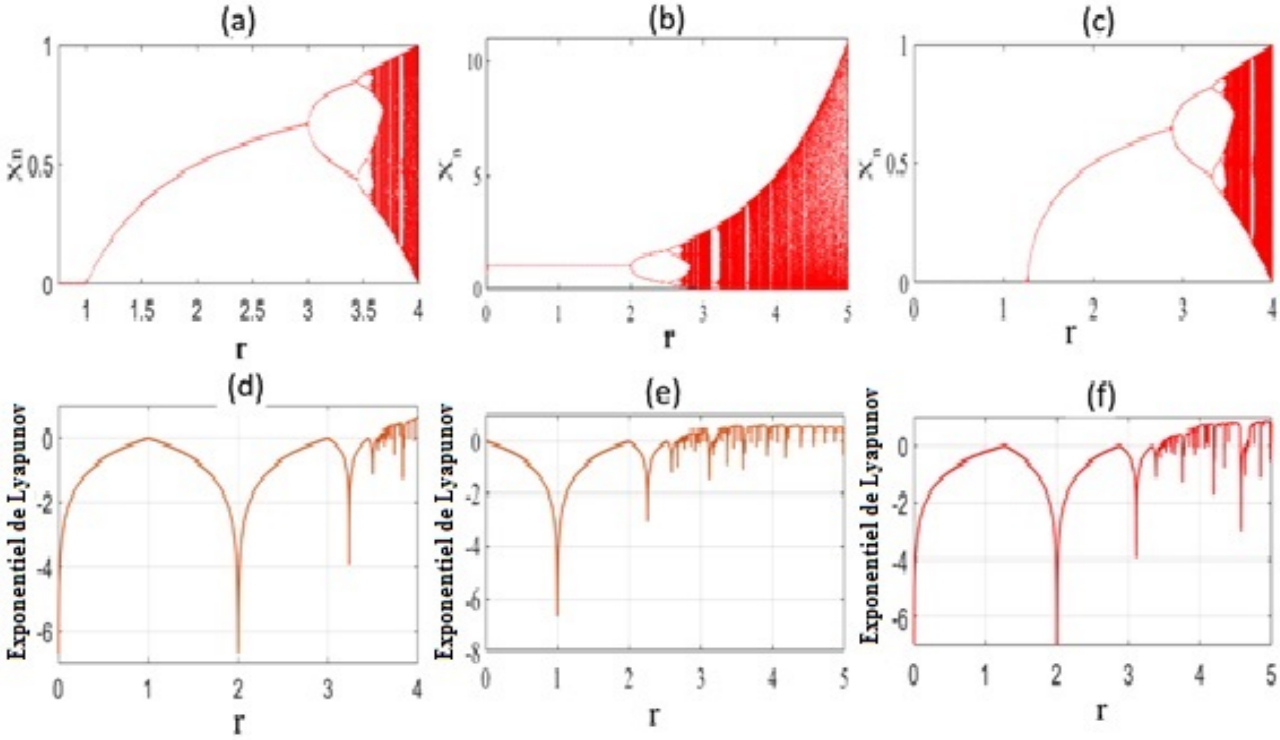


FIGURE 2.8 – Diagrammes de bifurcation et graphe des exposants de Lyapunov. (a)-(c) Diagramme de bifurcation des cartes Logistique, May et Sine , (d)-(f) Exposants de Lyapunov des cartes Logistique, May et Sine.

Les cartes 1D telles que la logistique, May, Logistique, Sine se sont avérées peu appropriées pour construire des cryptosystèmes robustes à cause de certaines faiblesses telles que : un faible

espace de clé, les valeurs périodiques des données de sortie, les mauvaises propriétés ergodiques pour certaines gammes du paramètre de contrôle [6]. Pour résoudre ce problème, Zhou et al. [89] ont proposé de combiner les paramètres de ces différentes cartes. La figure 2.9 montre la nouvelle carte obtenue après combinaison de deux cartes chaotiques 1D.

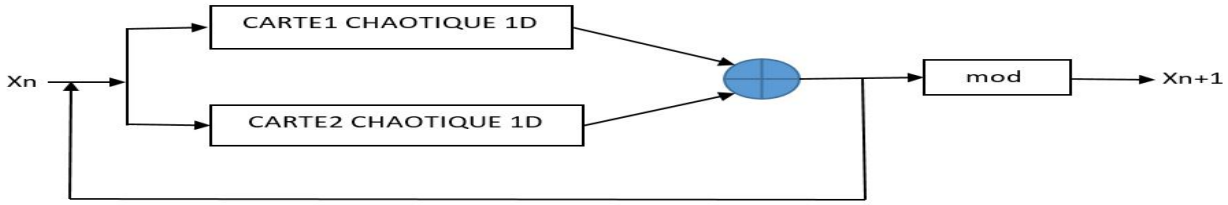


FIGURE 2.9 – Structure de la nouvelle carte.

D'après le modèle de la figure 2.9, nous combinons respectivement les cartes Logistique et May, puis Logistique et Sine afin d'obtenir deux cartes ayant des proriétés ameliorées , en comparaison avec les cartes originales. Ainsi, les cartes Logistique-May (LM) et Logistique-Sine (LS) sont définies respectivement par les relations 2.9 et 2.10 suivantes ([89], [90]) :

LM :

$$x_{n+1} = (x_n \exp((r+9)(1-x_n)) - (r+5)x_n(1-x_n)) \bmod 1 \tag{2.9}$$

Où $x_n \in [0,1]$ et $r \in [0,5]$.

LS :

$$x_{n+1} = (r x_n (1-x_n) + (4-r)\sin(\pi x_n)/4) \bmod 1 \tag{2.10}$$

Où $x_n \in [0,1]$ et $r \in [0,4]$.

La figure 2.10 présente le diagramme de bifurcation et l'exposant de Lyapunov des cartes Logistique-May (LM) et Logistique-Sine (LS) respectivement. Nous pouvons apprécier sur le schéma le fait que les propriétés chaotiques sont excellentes dans l'intervalle , et la valeur maximale de l'exposant de Lyapunov est égale à 8,3 pour LM et 0,7 pour LS. Avec cette valeur élevée de l'exposant de Lyapunov, la carte Logistique-May est mieux adaptée pour sécuriser les données et augmenter le temps de chiffrement si la structure algébrique de l'algorithme de cryptage est bonne.

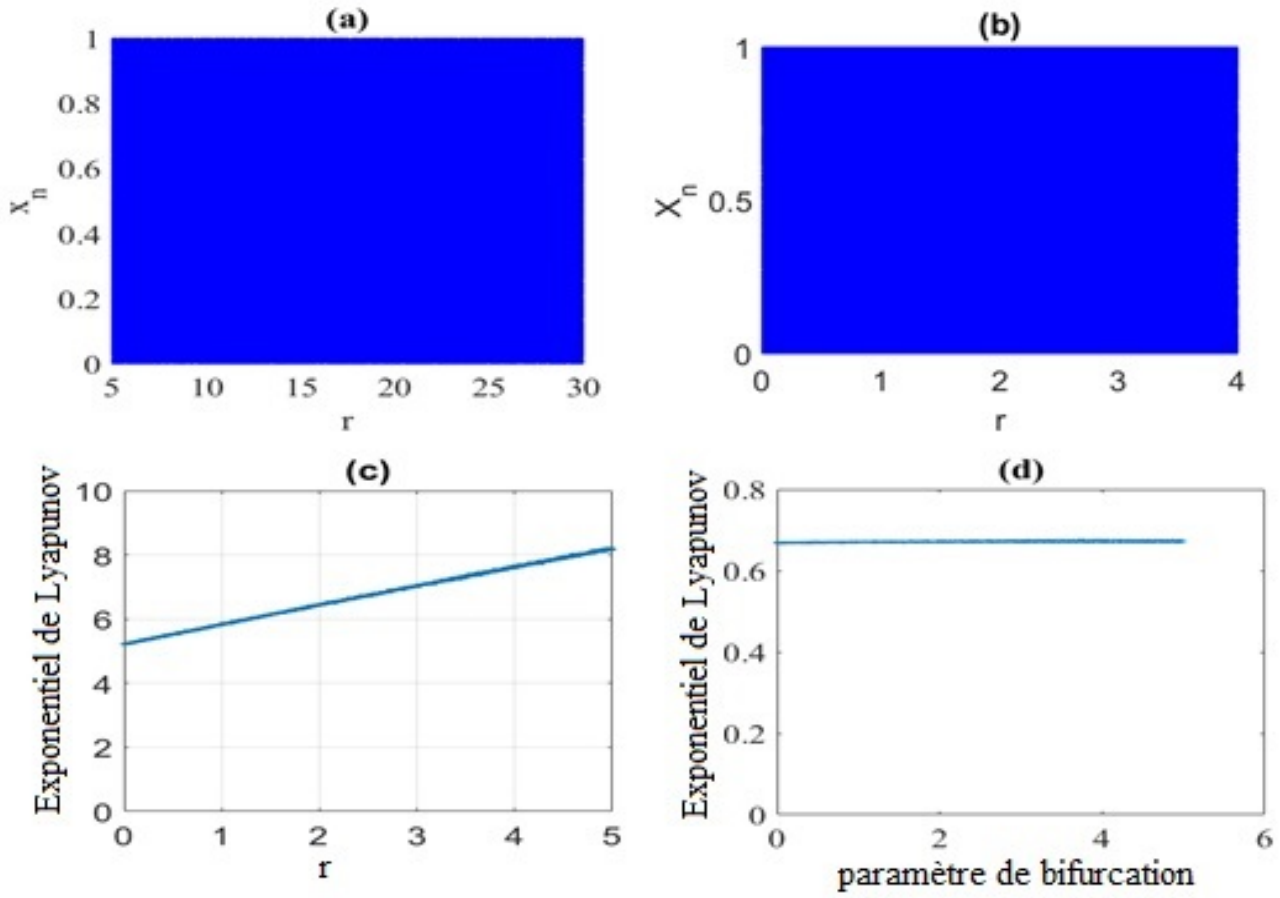


FIGURE 2.10 – Cartes Logistique-May (LM) et Logistique-Sine (LS) : (a)-(b) Diagrammes de Bifurcation de LM et LS, (c)-(d) Exposants de Lyaponuv de (LM) et (LS).

2.4.1.3 Carte linéaire chaotique par morceaux

La fonction linéaire chaotique par morceaux (en anglais PWLCM : Piece Wise Linear Chaotic Map) est un système dynamique à une dimension principalement utilisée dans les cryptosystèmes pour sa simplicité, son efficacité d'exécution et présente une bonne densité pour toutes les valeurs du paramètre de contrôle. Elle est définie par la relation 2.11 :

$$u_{i+1} = \begin{cases} u_i/p & 0 \leq u_i < p \\ (u_i - p)/(0.5 - p) & p \leq u_i < 0.5 \\ (1 - u_i) & u_i \geq 0.5 \end{cases} \tag{2.11}$$

Où $u_i \in [0, 1]$, et $p \in [0, 0.5]$ est le paramètre de contrôle.
La figure 2.11 illustre l'évolution temporelle de la fonction PWLCM et son diagramme de bifurcation.

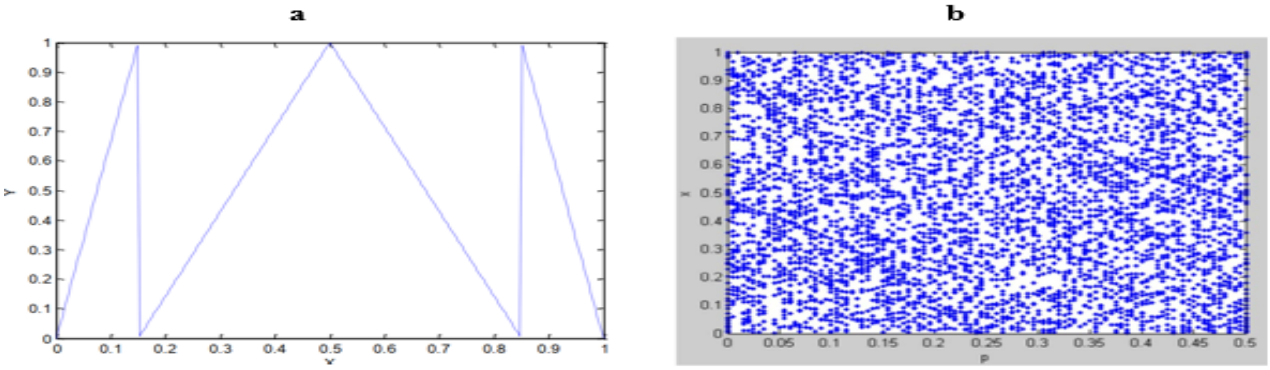


FIGURE 2.11 – Carte PWLCM : (a) évolution temporelle, (b) diagramme de bifurcation.

2.4.2 La Transformée en Cosinus Discrète (DCT)

La Transformation en Cosinus Discrète est une variante de la transformée de Fourier. Elle prend pour une image par exemple, un ensemble de points du domaine spatial et le transforme en une représentation identique du domaine des fréquences. Cette transformation est réversible, et son inverse (IDCT) consiste à passer de la représentation fréquentielle à celle spatiale. L'une des propriétés importantes de la DCT est la possibilité de regrouper l'essentiel de l'information dans le coin gauche supérieur de la représentation spectrale obtenue. Cette propriété permet de fusionner plusieurs spectres d'images en un seul spectre afin d'effectuer une transmission simultanée.

Les équations 2.12 et 2.13 suivantes définissent respectivement la DCT et la IDCT :

$$y_{k,l} = \frac{c(k).c(l)}{4} \sum_{n=0}^{7} \sum_{m=0}^{7} x_{n,m} \cos\left(\frac{(2n+1)k\pi}{16}\right) \cos\left(\frac{(2m+1)l\pi}{16}\right) \tag{2.12}$$

$$x_{n,m} = \frac{1}{4} \sum_{k=0}^{7} \sum_{l=0}^{7} c(k)c(l)y_{k,l} \cos\left(\frac{(2k+1)n\pi}{16}\right) \cos\left(\frac{(2l+1)m\pi}{16}\right) \tag{2.13}$$

Avec $c(\alpha) = \begin{cases} \frac{1}{\sqrt{2}} & pour \quad \alpha = 0 \\ 1 & pour \quad \alpha \neq 0 \end{cases}$

Dans la pratique, la DCT éclate une zone d'image en fréquences à deux dimensions. La plus basse d'entre elles est placée dans le coin supérieur gauche. Les fréquences horizontales croissent vers la droite et les fréquences verticales croissent vers le bas. La DCT est une méthode de compression avec pertes, bien qu'elles soient moindres. Ces pertes sont dues aux erreurs d'arrondi lors des calculs des coefficients de la DCT ; toutefois, l'image reconstituée après transformation est toute aussi proche de l'image originale.

2.4.3 Filtrage fréquentiel

Le filtrage fréquentiel est un procédé utilisé comme outil de fusion de plusieurs images de référence. Il procède au multiplexage des images cibles par différents moyens en fonction de l'application désirée. Un exemple est le filtre composite défini par la relation 2.14.

$$H_{composite} = \sum \alpha_i R_i \tag{2.14}$$

Le principe de ce filtre consiste à former une combinaison de différentes références pondérées par des constantes afin d'optimiser une fonction de coût choisie pour une application donnée. Chaque pixel du plan de filtre est la somme des valeurs issues des différents spectres d'images références considérées. Ce filtre est très efficace, mais lorsque le nombre de références augmente, ce filtre présente un phénomène de saturation qui entraîne une baisse significative de ses performances.
Dans le souci de corriger ce problème de saturation, Alfalou et al. [59] ont proposé une version optimisée du filtre composite appelée le filtre segmenté. Son principe consiste à diviser le plan de Fourier du filtre en plusieurs zones. Chacune de ces zones sera allouée à un spectre d'une des images de référence utilisées pour la construction du filtre (Figure 2.12). Le problème majeur posé par ce filtre est celui des pixels isolés lorsque les images à multiplexer augmentent. En effet, il devient difficile de reconstituer les images de référence dans leur intégralité, car peu de pixels consécutifs appartiennent à la même référence. Toutefois, ce filtre présente de meilleures performances comparées au filtre composite.

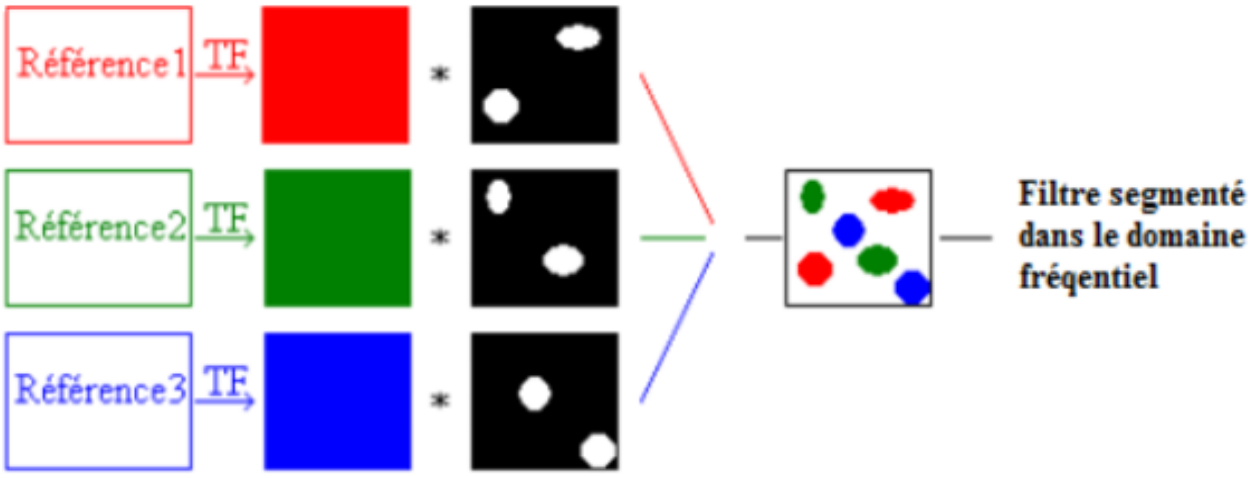


FIGURE 2.12 – Principe de construction d'un filtre segmenté [59]

Dans le cadre de ce travail, nous utiliserons dans le deuxième algorithme développé les atouts du filtrage fréquentiel ainsi que la transformation Discrète en Cosinus pour effectuer la fusion des images de références. La DCT présente l'avantage de regrouper dans le coin gauche du plan spectral de l'image l'essentiel de l'information. Ainsi, il devient possible de fusionner plusieurs images en un spectre et les reconstituer aisément sans grande perte d'informations.

2.4.4 Sous-blocs d'images

La théorie des matrices peur être utilisée pour subdiviser une grande matrice en plusieurs petites matrices, et vice versa. Une image étant une matrice, dans le domaine de traitement d'images, il est évident de subdiviser une image en sous-blocs d'images, et faire la procédure inverse. Seulement, lorsque les sous d'images sont de petite taille, cela participe de la décorrélation de l'image originale. En plus, si les petites images sont permutées, les pixels de l'image deviennent fortement décorrélés, ce qui est avantageux en cryptographie.
Supposons que $O1_{m\times n}, O2_{m\times n}, ..., Ok_{m\times n}$ sont k images originales. $O1_{m\times n}$ peut être subdivisé en un ensemble de petites images, $B1_i$ Chaque élément $B1_i \in \{B1_i\}$ est appelé élément d'image pure.
D'autres part, k ensembles d'éléments d'images purs $\{B1_i\}, \{B2_i\}, ..., \{Bk_i\}$ peuvent être crées à partir des images $O1_{m\times n}, O2_{m\times n}, ..., Ok_{m\times n}$ respectivement. Un grand ensemble
$C = \{B1_i\} \cup \{B2_i\} \cup ... \cup \{Bk_i\}$ peut être obtenu en mixant ensemble les éléments d'images pures. Chaque élément $C_i \in \{C_i\}$ est appelé élément d'image mixé. L'image mixée C contient toutes les informations des images originales $Oi_{m\times n}$ et présente une structure désorganisée (décorrélée de pixels). Par exemple la figure 2.13 présente une image composite et son image mixée correspondante. Ici, le mixage est effectué par une procédure de permutation des blocs d'images.

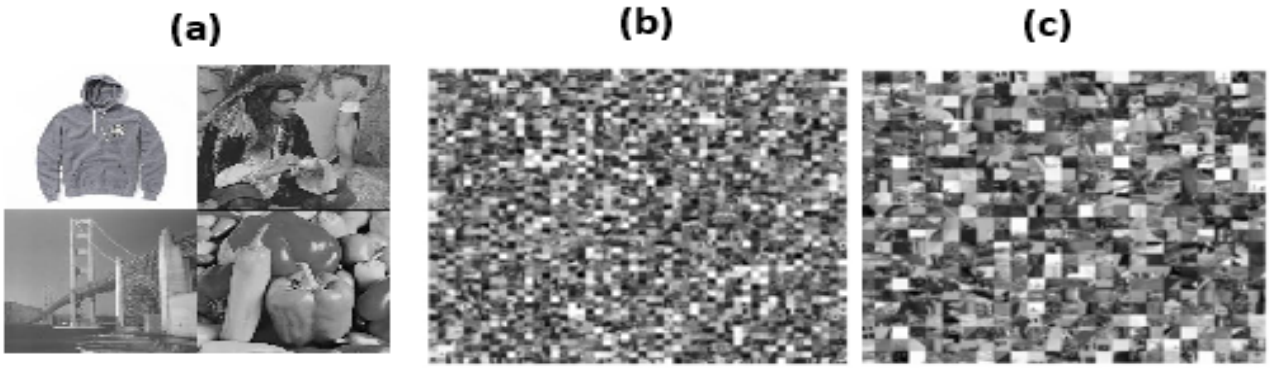


FIGURE 2.13 – Image originale composite et images mixées. (a) image composite, (b) blocs mixés de taille 16×16 , (c) blocs mixés de taille 32×32.

2.5 Contribution 1 : Transmission sécurisée d'images médicales pour la télémédecine

2.5.1 Description

Ces dernières années, la télémédecine a connu un essor considérable grâce au développement des technologies de l'information et de la communication. De nombreuses informations médicales ont ainsi été échangées entre les praticiens de la santé. Il est à noter qu'une grande partie de ces informations est constituée d'images médicales. La plupart d'entre elles contiennent des informations sensibles sur les patients, et doivent donc rester confidentielles. Dans le souci d'apporter une réponse à ce besoin, nous proposons un algorithme de cryptage de plusieurs images médicales de différents types en utilisant une approche hybride, et dont la transmission est effectuée à travers un canal de transmission non sécurisé. Dans le processus de cryptage, deux images sont d'abord brouillées séparément avec des cartes chaotiques, puis fusionnées pour améliorer la robustesse du cryptosystème et élargir son espace clé. La première image est chiffrée par les processus de permutation et de diffusion en utilisant respectivement les cartes Logistique-May (LM) et Henon. Les pixels de la seconde image sont brouillés par le système Logistique-Sine (LS). Enfin, la fusion des deux images chiffrées est réalisée par une expression mathématique non linéaire basée sur la règle de Cramer pour obtenir deux images hybrides chiffrées.

Le cryptosystème proposé présente plusieurs atouts :

- **La sécurité des données transmises est assurée à deux niveaux** : le cryptosystème proposé assure la sécurité des données transmises au niveau conventionnel, via la clé de cryptage, et au niveau du canal de transmission. Pour le second cas, l'image cryptée n'est pas cohérente, car elle contient des informations provenant d'autres images sources grâce à la technique innovante de cryptage hybride utilisée. De ce fait, la nature de l'image chiffrée constitue un niveau de sécurité dans le canal ;

- **Le mécanisme de transmission du cryptosystème est complexe** : tel que présenté à la figure 2.14, les données médicales du patient sont d'abord cryptées, puis envoyées par un modem depuis le système d'émission vers le système de réception à travers le canal de transmission. Ainsi, le dispositif numérique associé (système d'émission, modem, système de réception) constitue un premier niveau de sécurité du cryptosystème. Le deuxième niveau de sécurité réside au niveau du protocole de chiffrement hybride adopté dans l'algorithme, car cela assure la transmission sécurisée des données dans le cala, ce qui augmente l'espace de la clé et rend très complexe toute attaque éventuelle.

- **La sécurité du cryptosystème est renforcée grâce à l'utilisation d'une combinaison des cartes 1D logistiques, May et Sine** : Le niveau de sécurité d'un système cryptographique basé sur le chaos dépend de la bonne qualité de la carte chaotique utilisée. Les cartes 1D combinées (cartes Logistique-May et Logistique-Sine), utilisées dans ce travail, présentent de meilleures propriétés chaotiques par rapport à leurs sources originales, et conviennent au cryptage des images médicales.

- **La transmission des données présente un haut niveau de sécurité** : Les images hybrides cryptées obtenues pour la transmission peuvent être envoyées au récepteur par des canaux indépendants et différents. Ce procédé limite fortement l'action d'un attaquant qui doit intercepter toutes les images cryptées avant d'effectuer ses tests de cryptanalyse.

- **Le cryptosystème est simple et efficient** : la complexité du cryptosystème proposé est faible, car reposant principalement sur les opérations de permutations et de diffusion ; également, le temps de chiffrement des données est réduit et permet au cryptosystème d'être adapté aux applications en temps réel.

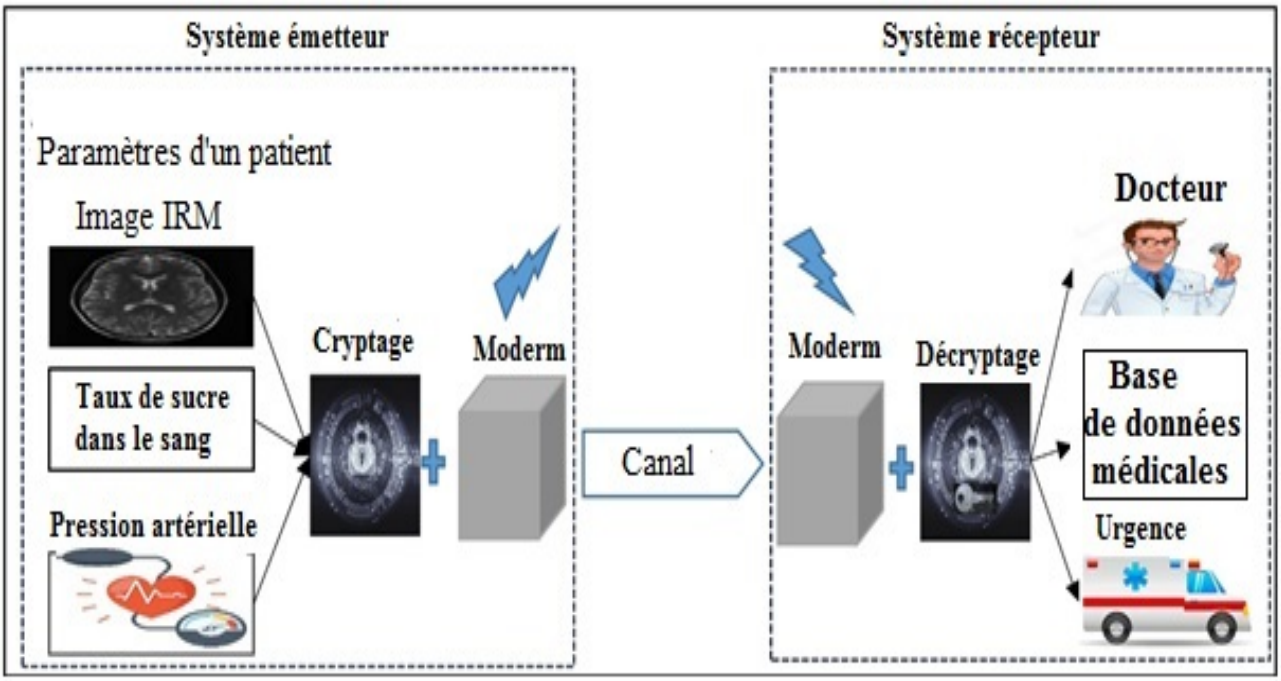


FIGURE 2.14 – Schéma block d'un système de télémédecine

2.5.2 Motivation du choix des méthodes de permutation et diffusion

Bon nombre d'algorithmes robustes obéissent à la structure permutation-diffusion afin de respecter les critères de Shannon. Certains d'entre eux chiffrent l'image en plusieurs tours selon la structure de Feistel (Figure 2.15), ce qui contribue à la robustesse du système, mais augmente tout de même le temps de cryptage. Afin de minimiser le temps de cryptage, de nouvelles méthodes de permutation, voire diffusion sont en quette de développement.

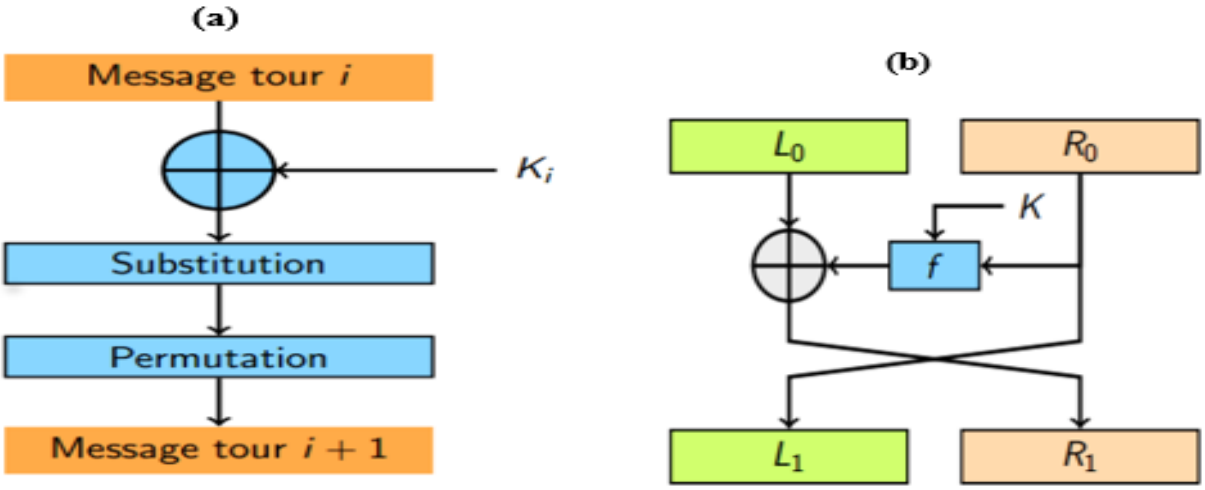


FIGURE 2.15 – Permutation à plusieurs tours et schéma de Feistel. (a) Permutation à plusieurs tours, (b) schéma de Feistel (f est la fonction de confusion, $L_1 = R_0, R_1 = L_0 \oplus f(R_0)$). Figure tirée de Bressson (2015).

∗ **La permutation**

La permutation consiste à changer la position des pixels d'une image sans changer leurs valeurs. Cette action a pour résultat une confusion des pixels entrainant une décorrélation de l'image. Les données provenant des générateurs de nombres pseudo-aléatoires (GNSA) sont généralement utilisées pour réaliser les opérations de confusion (Carte de Arnold, Tent, May, systèmes de Lorentz, Rösler, etc.). La permutation peut s'effectuer par plusieurs procédés : le brouillage, la substitution, l'utilisation des boites S (S-box : boîtes de permutations préconçues). Le niveau de sécurité de la permutation étant faible, elle devrait être associée à une méthode de diffusion pour rendre robuste le crypto-système ([60], [91]).

∗ **La diffusion**

La confusion vise à assurer une relation statistique plus complexe entre l'image et la clé de

cryptage. Dans ce cas, la valeur des pixels est modifiée sans changer leur position. Une approche plus simple d'effectuer la diffusion des pixels de l'image est d'effectuer un XOR entre les pixels et une matrice de valeurs aléatoires.

2.5.3 Algorithme de chiffrement de la première image

La première image (ou juxtaposition d'un ensemble d'images) est cryptée selon la structure permutation-diffusion. Le schéma de cryptage de l'algorithme utilisé est illustré à la figure 2.16 (a). Les principales étapes de l'algorithme sont :

- Choisir les conditions initiales du système Logistique-May, les rendant dépendantes de l'image cible ;
- Générer une séquence chaotique à partir du système logistique-May ;
- Brouiller l'image cible par un processus de permutation à partir de la séquence chaotique générée précédemment ;
- Effectuer la diffusion des pixels de l'image permutée en appliquant l'opérateur XOR entre cette dernière et la séquence chaotique générée par la carte de Henon.

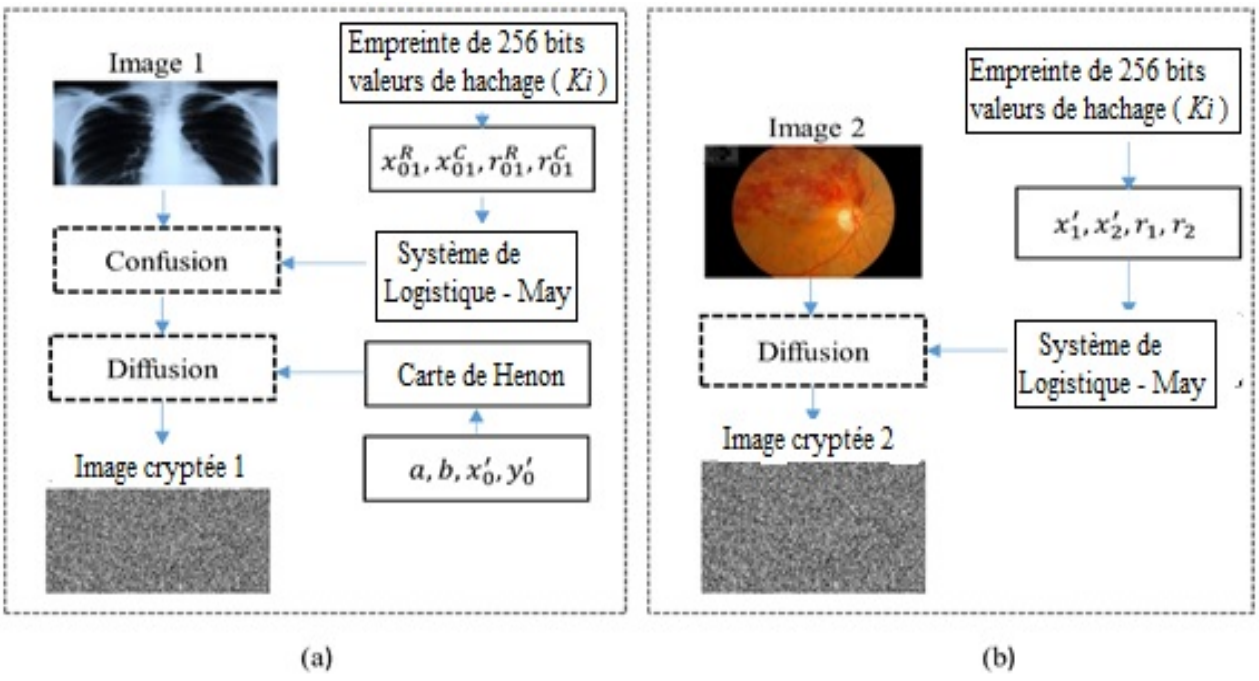


FIGURE 2.16 – Schéma de cryptage des images originales. (a) chiffrement de la première image, (b) chiffrement de la seconde image.

2.5.3.1 Processus de génération des clés.

Nous choisissons de générer les clés du cryptosystème constituées des conditions initiales et paramètres de contrôle des cartes utilisées en utilisant la fonction de hachage SHA- 256 pour assurer une bonne sensibilité de l'image aux petites variations des paramètres constituant la clé. La fonction SHA-256 produit d'une image une séquence unique de 256 bits. Deux images originales ayant un bit de différence produisent deux séquences de valeurs hachées différentes. Les étapes utilisées pour générer les clés sont les suivantes :

— Effectuer le haché de l'image originale (cible), on obtient une séquence de 256 bits dénotée k ;

— Diviser K en k_i blocks de 8 bits chacun ($i = 1, 2, \ldots, 32$) tel qu'indiqué par l'équation 2.15 suivante :

$$k = [k_1, k_2, ..., k_{32}] \tag{2.15}$$

— Calculer les paramètres $\left(x_{01}^R, x_{01}^C, r_{01}^R, r_{01}^C\right)$ utilisés pour chiffrer l'image originale I_1 suivant les équations (2.16) - (2.19) suivantes :

$$x_{01}^R = \frac{3}{4}x_0 + \frac{bin2dec\left(k_{17} \oplus k_{18} \oplus ... \oplus k_{24}\right)}{2^8 \times 10} \tag{2.16}$$

$$x_{01}^C = \frac{3}{4}x_0 + \frac{bin2dec\left(k_1 \oplus k_2 \oplus ... \oplus k_8\right)}{2^8 \times 10} \tag{2.17}$$

$$r_{01}^R = \frac{4}{5}r_0 + \frac{bin2dec\left(k_9 \oplus k_{10} \oplus ... \oplus k_{16}\right)}{2^8 \times 10} \tag{2.18}$$

$$r_{01}^C = \frac{4}{5}r_0 + \frac{bin2dec\left(k_{25} \oplus k_{26} \oplus ... \oplus k_{32}\right)}{2^8 \times 10} \tag{2.19}$$

Où $bin2dec\,(.)$ représente la fonction sous MATLAB pour convertir une séquence binaire en sa valeur décimale correspondante, le symbole $\oplus$ désigne l'opérateur OU-Exclusif ; $x_n \in [0, 0.9]$ et $r_0 \in [0, 4.9]$.

De manière similaire, les paramètres de la clé ($x_1^{'}$,$x_2^{'}$,r_1,r_2) utilisés pour chiffrer la deuxième image sont obtenus respectivement par les équations (2.16) - (2.19), avec K le haché de l'image I_2, $x_n \in [0, 0.9]$ et $r_0 \in [0, 3.9]$.

2.5.3.2 Processus de permutation

Soit I l'image cible de taille $M \times N$. Les étapes de la permutation sont les suivantes :

— Itérer l'équation 2.9 du système Logistique-May 100 fois (pour éviter l'effet d'artefact), puis N fois selon la direction des lignes, et M suivant celle des colonnes ;

— Ranger les valeurs de la séquence de nombres obtenus par ordre croissant ;
— Pour chaque position de couple ligne-colonne de pixels de l'image, chercher la position précédente du couple correspondant ligne-colonne des valeurs pseudo-aléatoires rangées par ordre croissant et les remplacer dans l'image ;
— Permuter la position du dernier pixel de l'image avec elle-même.

2.5.3.3 Étape de diffusion

Dans le souci d'assurer une relation statistique plus complexe entre l'image permutée et la clé de cryptage, nous utilisons la carte de Henon pour effectuer la confusion des pixels de l'image permutée. La diffusion dans son procédé modifie la valeur des pixels d'une image sans changer leur position. Les étapes de la phase de confusion sont :

— Fixer les paramètres , b et condition initiale $x_0^{'}$ et $y_0^{'}$ de la carte de Henon ;
— Convertir l'image permutée en un vecteur colonne ;
— Itérer l'équation (2.5) de Henon $M \times N$ fois et ranger les valeurs obtenues dans un matrice A de même taille que l'image permutée ;
— Disposer les valeurs de A suivant un vecteur colonne de longueur $M \times N$. Faire de même pour les pixels de l'image permutée ;
— Appliquer l'opérateur XOR entre l'image permutée et la matrice A contenant les valeurs pseudo-aléatoires obtenues de la carte de Henon, puis convertir les pixels de l'image obtenue sous le format 8bits pour obtenir l'image chiffrée.

2.5.4 Algorithme de chiffrement de la deuxième image.

Le cryptage de la deuxième image du cryptosystème est réalisé en utilisant une approche de diffusion. La carte Logistique-Sine (LS) est utilisée pour générer une séquence de nombres pseudo-aléatoires exploitée pour la transformation des pixels de l'image d'entrée. Le schéma du processus de chiffrement est illustré à la figure 2.16 (b), et les étapes principales sont les suivantes :

1. Fixer la condition initiale $x_1^{'}$ et le parameter de contrôle r_1 de la carte (LS) ;
2. Génerer une séquence de nombres pseudo-aléatoires $X = \{x_i\}$ de longueur $M \times N$ en itérant l'équation (2.10) du système LS ; avec $[M, N]$ les dimensions de l'image.
3. Convertir les valeurs précédentes en des nombres entiers en utilisant les relations (2.20) suivantes :

$$X = ceil\,(X \times 255) \tag{2.20}$$

   Où $ceil(x)$ est l'opérateur dans MATLAB qui arrondit l'élément x à l'entier qui lui est plus proche ;
4. Convertir les valeurs de X en une séquence binaire ;

5. Permuter de façon circulaire les valeurs binaires précédentes selon la direction des lignes en utilisant la relation (2.21) :

$$X = circshift\,(X, 1) \tag{2.21}$$

6. Convertir les valeurs de X obtenues en des valeurs décimales, puis transposer le vecteur X.
7. Utiliser les paramètres $x_2^{'}$ et r_2 pour générer une seconde séquence de nombres pseudo-aléatoires Y tel que décrit à l'étape 2 ;
8. Construire une séquence chaotique P pour diffuser les valeurs de pixels de l'image originale I suivant le relation (2.22) :

$$P = X \oplus Y \oplus I^{'} \tag{2.22}$$

   Où $I^{'}$ représente la matrice transposée de I.
9. Ajuster les éléments de P en une matrice de mêmes dimensions que celle de l'image originale I afin d'obtenir l'image chiffrée.

Le processus de décryptage est l'inverse de celui du cryptage.

2.5.5 Description du cryptosystème général

Dans cette section, les deux images cryptées respectivement par les algorithmes 1 et 2 vont être combinées afin d'obtenir deux images hybrides, beaucoup plus difficiles à reconstituer. Le schéma du cryptosystème général est illustré à la figure 2.17 ci-dessous.

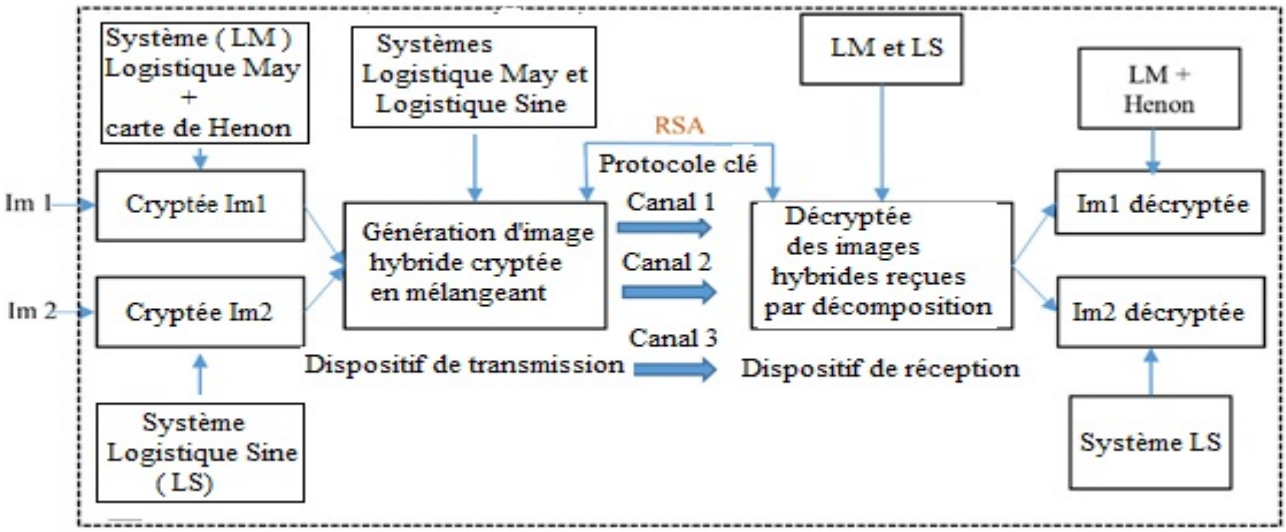


FIGURE 2.17 – Schéma de cryptage et décryptage hybride.

i) Processus de cryptage Dans le cryptosystème proposé, les images cryptées sont transmises simultanément par divers canaux non sécurisés. Les images 1 et image 2 cryptées séparément sont fusionnées pour produire deux images hybrides et incohérentes avant d'effectuer leur transmission au destinataire. Ce dernier devra alors être en possession des deux images envoyées par deux canaux différents et de la clé de cryptage pour déchiffrer les informations sources. Si les images à crypter sont supérieures à deux, il est possible de les regrouper en deux grandes images de même taille en entrée avant le chiffrement. Par exemple, la figure 2.18 montre comment quatre images médicales sont combinées en une image. Par ailleurs, s'il y'a une seule image à chiffrer, elle peut être subdivisée en deux sous-images de même taille, avant de procéder au chiffrement.

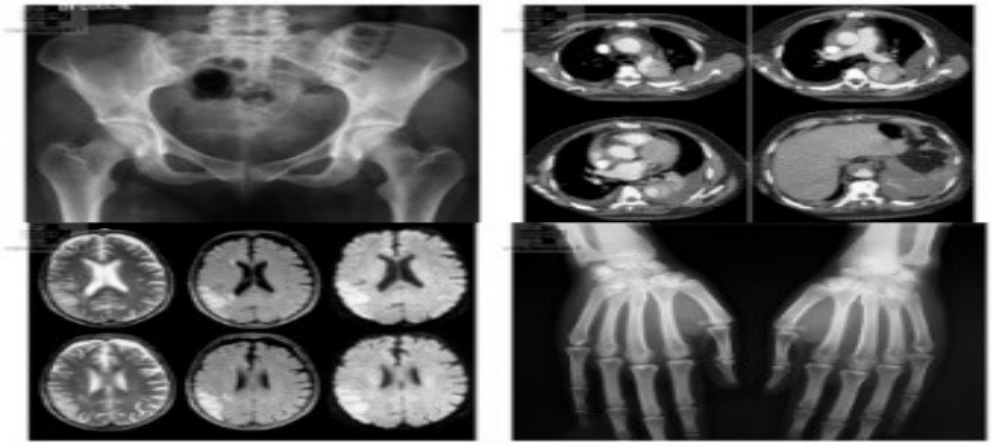


FIGURE 2.18 – Quatre images médicales combinées en une image

ii) Processus de fusion ou mixage

Les systèmes Logistique-May (LM) et Logistique-Sine (LS) sont utilisés pour générer deux séquences de nombres pseudo-aléatoires W_1 et W_2 après $2M \times 2N$ itérations chacune. Les valeurs obtenues sont additionnées pour former la séquence de valeurs chaotiques W suivant la relation (2.23) :

$$W = (W_1(i,j) + W_2(i,j)) \bmod 1 \tag{2.23}$$

Ainsi, les valeurs de W sont converties en des valeurs réelles sous le format de 8 bits suivant la relation (2.24) :

$$(W = u\text{int}8(W \times 255)) \tag{2.24}$$

Au cours de l'étape de fusion, les conditions initiales et paramètres de contrôle des cartes utilisées sont respectivement (x''_{01}, r''_1) pour (LM) et (x''_{02}, r''_2) pour (LS). Ces paramètres sont obtenus tel que décrit à la section 2.5.2.1 en considérant l'image d'entrée comme la somme des images 1 et 2 à l'entrée du cryptosystème. Pour obtenir les deux images cryptées, la matrice W est subdivisée en quatre sous matrices de taille $M \times N$ chacune tel qu'indiqué par la relation (2.25), puis sont combinées avec les deux images I_1 et I_2 cryptées séparément (Sections 2.5.2 et 2.5.3), suivant les équations (2.26) et (2.27) :

$$W = \begin{pmatrix} w_{11} & w_{12} \\ w_{21} & w_{22} \end{pmatrix} \tag{2.25}$$

$$C_1[i,j] = \left(w_{11} \times I_1[i,j] + w_{12} \times I_2[i,j]\right)_{\bmod 256} \oplus \left(floor\left(w_{11} \times w_{21}\right) \times 10^{15}\right) \tag{2.26}$$

$$C_2[i,j] = \left(w_{21} \times I_1[i,j] + w_{22} \times I_2[i,j]\right)_{\bmod 256} \oplus \left(floor\left(w_{12} \times w_{22}\right) \times 10^{15}\right) \tag{2.27}$$

Où $C_1[i,j]$ et $C_2[i,j]$ sont les images cryptées hybrides obtenues à la sortie du cryptosytème, le symbole $\oplus$ représente l'opérateur XOR (OU-Exclusif) appliqué entre les bits des pixels, $floor(x)$ est l'opérateur dans MATLAB qui arrondit la valeur de l'élément x à l'entier le plus proche. Le produit mixte $w_{ij} \times w_{ji}$ dans ces relations précédentes renforce la qualité des images fusionées.

iii) Processus de décryptage

Le processus de décryptage est illustré à la figure 2.19. Si le destinataire dispose des clés du cryptosystème, les deux images hybrides cryptées sont premièrement décomposées en utilisant la règle de Kramer pour résoudre le système d'équations suivant :

$$\begin{cases} \left(I_1[i,j] \times w_{11} + I_2[i,j] \times w_{12}\right)_{\bmod 256} = C_1 \oplus \left(floor\left(w_{11} \times w_{21}\right) \times 10^{15}\right) \\ \left(I_1[i,j] \times w_{21} + I_2[i,j] \times w_{22}\right)_{\bmod 256} = C_2 \oplus \left(floor\left(w_{12} \times w_{22}\right) \times 10^{15}\right) \end{cases} \tag{2.28}$$

La résolution de (2.28) permet de retrouver I_1 et I_2 . Les deux images originales d'entrée peuvent être retrouvées en décryptant respectivement I_1 et I_2 selon l'algorithme utilisé pour les chiffer.

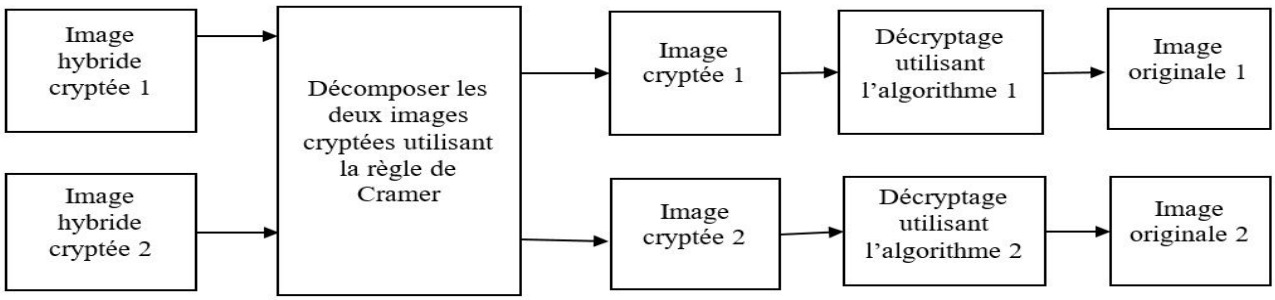


FIGURE 2.19 – Schéma de décryptage d'images hybrides

iv) Protocole d'échange de clés

Dans le cryptosystème, nous supposons que l'émetteur du message est nommé Alice et le destinataire Bob se sont accordés sur le protocole d'échange de clés. Ils peuvent par exemple choisir de crypter les paramètres de la clé par l'algorithme symétrique RS [65] tel que présenté sur le schéma général du cyptosystème (Figure 2.17). En conséquence, tous les paramètres du cryptosystème sont rangés dans un certain ordre afin de former un seul message M. Alice, à partir de sa clé publique va chiffrer le message M constitué des paramètres de la clé et l'envoyer par la suite à Bob ; ce dernier va se servir de sa clé privée pour rentrer en connaissance du la clé M du cryptosystème.

Comme illustration, la clé du cryptosystème proposé possède quatorze (14) paramètres de six chiffres chacun, tel qu'indiqué dans le tableau 2.1. Ainsi, le message M composé des éléments de la clé de cryptage est tel que : M =

$x_{01}^{R}\ x_{01}^{C}\ r_{01}^{C}\ a\ b\ x_0^{'}\ y_0^{'}\ x_1^{'}\ x_2^{'}\ r_1\ r_2\ x_{01}^{''}\ r_1^{''}\ x_{02}^{''}\ r_2^{''}$ = 099295098874499871498601139012023048054289012658097402096719397238395973088954499961091058392751

Alice va subdiviser le message M en 14 blocks (m_i) de six chiffres chacun, soit M = m_1 m_2,...,m_{14}, puis chiffrer M à l'aide de l'algorithme RSA ; ici, m_i correspond au ième paramètre dans M. Du coté du destinataire, dans le souci de déchiffrer M, Bob va reconstituer les 14 paramètres de la clé en considérant pour chaque paquet de m_i le bit de poids le plus fort comme la partie entière de la valeur, et les cinq autres bits comme partie imaginaire de la valeur du paramètre.

TABLE 2.1 – Paramètres de la clé

	Paramètres
Algorithme 1	$x_{01}^{R} = 0.99295$; $x_{01}^{C} = 0.98874$; $r_{01}^{R} = 4.99871$; $r_{01}^{C} = 4.98601$; $a = 1.39012$; $x_0^{'} = 0.54289$; $y_0^{'} = 0.12658$
Algorithme 2	$x_1^{'} = 0.97402$; $x_2^{'} = 0.96719$; $r_1 = 3.97238$; $r_2 = 3.95973$
Fusion	$x_{01}^{''} = 0.88954$;$r_1^{''} = 4.99961$;$x_{02}^{''} = 0.91058$; $r_2^{''} = 3.92751$

2.6 Contribution 2 : Algorithme de fusion et cryptage d'images utilisant la Transformation Discrète en Cosinus et les générateurs de nombres pseudo-aléatoires.

La Transformée Discrète en Cosinus présente un atout d'effectuer la fusion spectrale de plusieurs images, tout en conservant une bonne qualité des images reconstruites lorsque le nombre d'images à fusionner n'est pas important. Dans le souci d'assurer la transmission de

plusieurs images sans toutefois compromettre la qualité des images décryptées, nous proposons dans cette partie un algorithme approprié.

2.6.1 Description de l'algorithme

L'algorithme proposé repose sur la structure permutation-diffusion, assurée par les cartes chaotiques. Nous utilisons la transformée Discrète en Cosinus (DCT) et un filtre passe bas de taille appropriée pour effectuer la fusion spectrale des images à multiplexer en deux spectres d'images de même taille. Le fait de grouper les images cibles en deux spectres maximise le nombre d'images à transmettre et réduit l'erreur obtenue sur la qualité d'images décryptées. Ensuite, chacun de ces spectres est divisé en blocks de petite taille qui sont mixés entre eux par un processus de permutation. Afin d'obtenir une image cryptée robuste, la fusion des deux images brouillées précédentes est assurée par une relation non-linéaire basée sur la règle de Kramer. Les nouvelles cartes chaotiques développées par les auteurs Kamdeu Y. et al. [90] sont utilisés comme générateurs de nombres pseudo-aléatoires (GNPA) dans le processus de chiffrement. Il s'agit des systèmes 1D May-Gaussienne et Gaussienne-Gompertz obtenus après fusion des cartes May, Gaussienne et Gompertz. A la fin du processus, les deux images cryptées obtenues sont transmises au destinataire de manière sécurisée, ce qui diminue la quantité de données à transmettre. Le schéma général illustrant le processus de chiffrement est illustré par la figure 2.20.

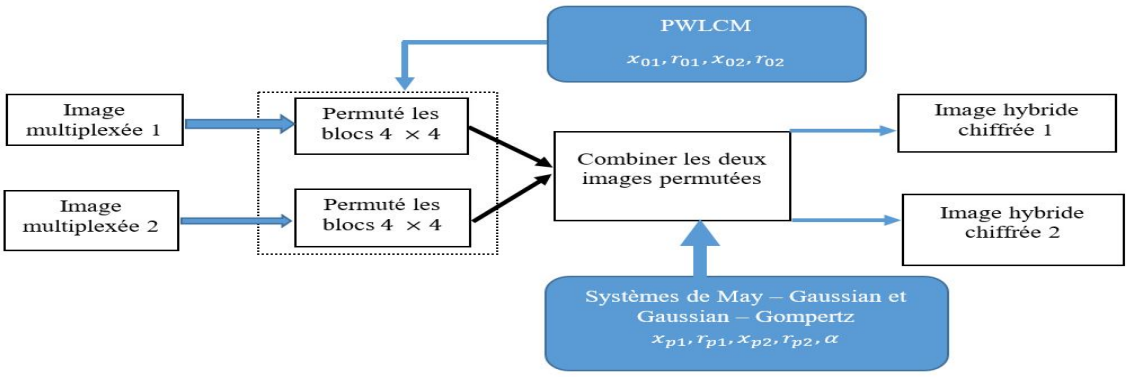


FIGURE 2.20 – Schéma de chiffrement utilisant la DCT et les GNPA

2.6.2 Fusion spectrale d'images cibles

Nous considérons N images cibles à multiplexer de taille , qui sont combinées en deux images M_1 et M_2 contenant chacune $N/2$ images cibles. Pour combiner les images cibles en M_1 et M_2 respectivement, la Transformée Discrète en Cosinus est appliquée séparément sur chaque image

cible. Ensuite, chaque spectre est multiplié par un filtre passe bas, de taille $(M^{'}, M^{'})$. Cette opération permet de prendre la partie essentielle de chaque spectre d'image cible nécessaire pour sa reconstitution sans la dégrader qualitativement. Le rapport de compression après multiplexage est défini par la relation 2.35 :
$C_r = 1 - (size\,of\,multiplexed\,DCT\,spetral\,plane)/size\,of\,N\,inputs\,images$

$$C_r = 1 - \left(M^2/N \times M^2\right) = 1 - (1/N) \tag{2.29}$$

Les images M_1 et M_2 regroupant les différents spectres sont obtenues après rotation et simple addition de ces derniers. En effet, les différents spectres sont décalés pour éviter leur chevauchement dans l'image fusionnée. La transformation inverse DCT est appliquée sur chacune des images M_1 et M_2 après fusion des différents spectres.

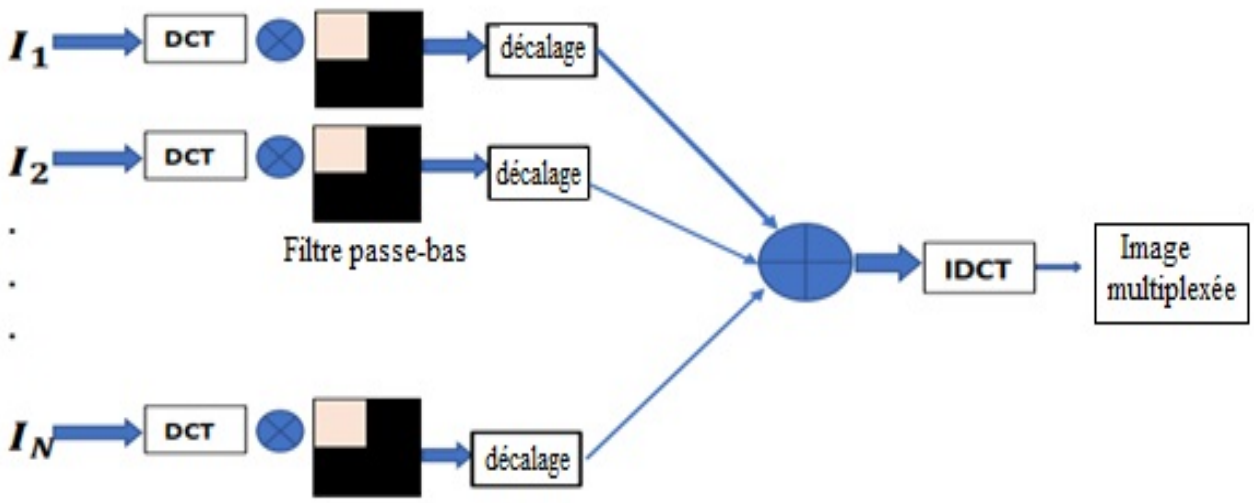


FIGURE 2.21 – Fusion spectrale des images cibles

La figure 2.21 illustre le processus de fusion des images cibles. Il est à préciser que le nombre d'images à multiplexer peut-être augmenté en ajustant la taille $(M^{'}, M^{'})$ du filtre passe bas. Plus le filtre est petit, plus le nombre d'images à multiplexer est grand, mais cela dégrade la qualité des images reconstruites. Nous avons choisi de regrouper les images cibles en deux images mixtes pour maximiser le nombre d'images à fusionner et obtenir une bonne qualité d'images reconstruites.

2.6.3 Permutation des blocs d'images

Chacune des images multiplexée M_1 et M_2 est considérée comme image d'entrée, et est subdivisée en blocks de petite taille 4×4 . En choisissant les blocks de petite taille, cela entraine

une décorrélation des pixels de l'image et une augmentation de l'entropie. Les différentes étapes utilisées pour permuter les blocks de M_1 ou M_2 sont décrites comme suit :

- Diviser l'image I de taille $M \times M$ en k blocks de taille 4×4 , avec $k = \frac{M}{4} \times \frac{M}{4}$.
- À partir des conditions initiales et paramètres de contrôle x_{01}, p_1 de la fonction PWLCM, générer une séquence de nombres en itérant k fois l'équation 2.13. Les valeurs obtenues sont rangées dans un vecteur P de taille.
- Répéter l'étape 2 pour générer une nouvelle séquence de nombres en utilisant les nouvelles valeurs de condition initiale et paramètre de contrôle x_{02} et p_2.
- Ordonner les valeurs de la séquence chaotique P et obtenir une nouvelle séquence $P^{'}$ tel que :

$$P^{'} = \left\{ p_i^{'} \right\}_k = \left\{ p_{t_1}^{'}, p_{t_2}^{'}, ..., p_{t_k}^{'} \right\} \tag{2.30}$$

  Ainsi, $\{t_1, t_2, ..., t_k\}$ est la permutation de la séquence $1, 2, \ldots, k$.
- Numéroter dans un ordre croissant les blocks d'images obtenus à l'étape 1, et ajuster

FIGURE 2.22 – (a) Image Lena (256×256) subdivisée en blocks de taille (16×16), (b) blocks d'images permutées.

leur position en fonction de l'ordre de permutation défini à l'étape 4. L'image obtenue est une image décorrélée. Un exemple d'image dont les blocks ont été permutés est illustré à la figure 2.22.

Les paramètres de clés x_{01}, p_1, x_{02} et p_2 utilisés pour assurer la permutation des blocs d'images sont déterminés par les relations

$$x_{0i} = \left(x_0 + mean(I_i)/255\right)_{\bmod 1} \tag{2.31}$$

$$p_i = p_0 + 0.1 \times \max\left(S_1, S_2\right)/N \times M \times 2^9 \tag{2.32}$$

Où l'image originale I est subdivisée en deux parties P_1 et P_2 ; S_1 est la somme des pixels de

P_1 et S_2 pour P_2 ; $x_0 \in [0, 0.9]$, $p_0 \in [0, 0.4]$.

2.6.4 Fusion des spectres permutés

Les deux images M_1 ou M_2 regroupant les spectres sont fusionnées en deux images mixées. Au cours du processus, deux séquences chaotiques sont générées après $2M \times 2M$ itérations à partir des systèmes May-Gaussian et Gaussian-Gompertz définies par les équations 2.33 et 2.34 [90]. Les valeurs obtenues sont rangées en deux matrices W et T de taille $2M \times 2M$ respectivement.

- Système May-Gaussian

$$x_{n+1} = \left(\left(x_n \exp\left(r + 10\right)\left(1 - x_n\right)\right) + \frac{(r+5)}{4} + \exp\left(-\alpha x_n^2\right) \right) \bmod 2 \tag{2.33}$$

Avec $x_n \in [0, 1]$, $r \in [0, 5]$, $\alpha \in [4.7, 17]$. La valeur de l'exposant de Lyapunov du système est située dans la plage [2.5, 5.6].

- Système Gaussian-Gompertz

$$x_{n+1} = \left(\frac{(r/5 + 26)}{4} + \exp\left(-\alpha x_n^2\right) - (r/5 + 26)\, x_n \log\, x_n \right) \bmod 2 \tag{2.34}$$

Avec $x_n \in [0, 1]$, $r \in [0, 5]$, $\alpha \in [4.7, 17]$. La valeur de l'exposant du système est égale à 2.5. Les conditions initiales et paramètres de contrôle x_{p1} , r_{p1}, x_{p2}, r_{p2} des deux systèmes précédents utilisés pour le processus de fusion sont définis par les relations suivantes :

$$x_{pi} = (x_0 + 0.1 \times mean(I_i)/256) \tag{2.35}$$

$$r_{pi} = r + 0.1 \times [\min(I_i + 1)/\max(I_i + 2)] \tag{2.36}$$

Où $mean(I_i)$ représente la valeur moyenne des pixels de chacune des deux images I_i , $(i = 1, 2)$ regroupant les images cibles ; $max(I_i)$ et $min(I_i)$ représentent respectivement la valeur maximale et minimale du pixel de l'image I_i ,$(i = 1, 2)$; $x_0 \in [0, 0.9]$, $r \in [0, 4.9]$.
Les matrices W et T sont subdivisés en 4 blocs et combinées aux images permutées I_1 et I_2 de façon linéaire par les relations (2.37) et (2.38) :

$$C_1[i, j] = \left[\left(w_{11} \times I_1[i, j] + w_{12} \times I_2[i, j]\right)_{\bmod 256} \oplus floor(t_{11} \times t_{21}) \times 10^{15} \right] \tag{2.37}$$

$$C_2[i, j] = \left[\left(w_{21} \times I_1[i, j] + w_{22} \times I_2[i, j]\right)_{\bmod 256} \oplus floor(t_{12} \times t_{22}) \times 10^{15} \right] \tag{2.38}$$

Avec $W = \begin{pmatrix} w_{11} & w_{12} \\ w_{21} & w_{22} \end{pmatrix}$; $T = \begin{pmatrix} t_{11} & t_{12} \\ t_{21} & t_{22} \end{pmatrix}$

$C_1[i,j]$ et $C_2[i,j]$ sont les deux images cryptées obtenues à la fin du processus ; le symbole représente l'opérateur logique XOR (OU Exclusif). Le produit mixte $t_{ij} \times t_{ji}$, ainsi que l'utilisation de deux générateurs de nombres contribue à renforcer la robustesse du système et à élargir l'espace de clés.

∗ **Processus de décryptage**

Ayant connaissance des clés utilisées à l'étape de fusion $(x_{p1}, r_{p1}, x_{p2}, r_{p2}, \alpha)$, les deux images I_1 et I_2 regroupant les différents spectres d'images cibles peuvent retrouvées en résolvant le système d'équations suivant :

$$\begin{cases} \left(I_1[i,j] \times w_{11} + I_2[i,j] \times w_{12}\right)_{\bmod 256} = C_1\left(floor\left(t_{11} \times t_{21}\right) \times 10^{15}\right) \\ \left(I_1[i,j] \times w_{11} + I_2[i,j] \times w_{22}\right)_{\bmod 256} = C_2\left(floor\left(t_{12} \times t_{22}\right) \times 10^{15}\right) \end{cases} \tag{2.39}$$

Les différentes images cibles originales peuvent être obtenues après application sur I_1 et I_2 de l'opération inverse de permutation, de la Transformée Discrète en Cosinus ; puis la transformation inverse de DCT est appliquée à chaque spectre d'image cible.

2.7 Outils d'évaluation des algorithmes de cryptage d'images

Plusieurs algorithmes de chiffrement d'images ne cessent d'être développés par les chercheurs pour satisfaire le besoin de transmission sécurisée des informations. Cependant, tous ne parviennent pas à résister aux différentes attaques menées par les pirates. De ce fait, pour apprécier les performances d'un algorithme de chiffrement, des métriques portant tant sur l'analyse statistique que différentielle sont appliquées à l'algorithme de chiffrement. Dans le paragraphe qui suit, nous décrirons les principaux outils et métriques d'évaluation.

2.7.1 Outils d'analyse Statistique

Les outils d'analyse statistique d'images sont nécessaires pour confirmer l'efficacité d'une technique de chiffrage d'image. Les différentes caractéristiques d'un algorithme de chiffrage d'images sont explorées en utilisant plusieurs paramètres dont les principaux sont énumérés en dessous.

2.7.1.1 Analyse d'histogramme

L'histogramme d'une image est une métrique qui révèle la distribution des différents niveaux de gris des pixels de l'image. A la suite d'un processus de chiffrement sur une image,

l'histogramme de l'image résultante doit être totalement différent de celui de l'image originale. Généralement, l'histogramme de l'image chiffrée est uniforme (les pixels sont répartis de manière égale dans l'espace) si l'algorithme de cryptage utilisé est robuste. Un exemple d'histogramme d'une image chiffrée avec un algorithme robuste est illustré à la figure 2.23.

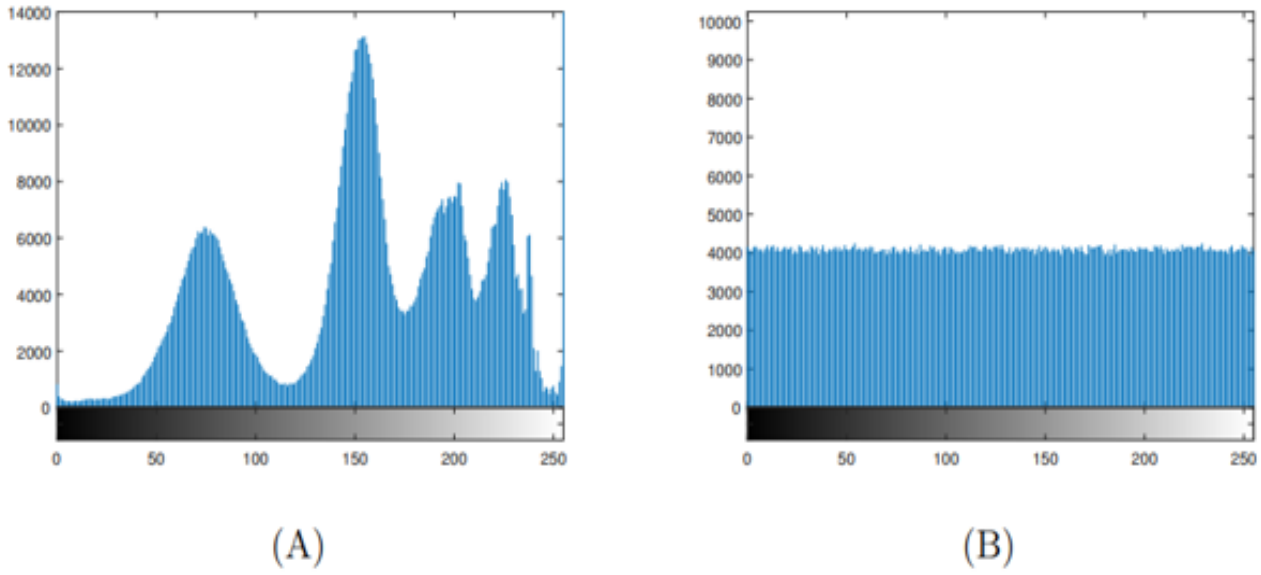


FIGURE 2.23 – (A) histogramme de l'image avant chiffrement, (B) histogramme de l'image après chiffrement.

2.7.1.2 Analyse par corrélation

Dans un cryptosystème, il est nécessaire d'effectuer les tests de corrélation entre les pixels voisins de l'image chiffrée afin de vérifier une éventuelle similarité avec les pixels correspondants de l'image originale. Ce test permet ainsi d'éviter l'attaque statistique sur le cryptosystème. Les valeurs des coefficients de corrélation des pixels adjacents d'une image suivant les trois directions (horizontale, verticale et diagonale) se situent entre -1 et +1 ; celles de l'image originale se rapprochent de 1 tandis que celles d'une image chiffrée avec un algorithme robuste se rapprochent de zéros. Le Coefficient de corrélation suivant une direction est calculé par la relation suivante :

$$C_c = \frac{M \times \sum_{i=1}^{M} X_i Y_i - \sum_{i=1}^{M} X_i^2 \times \sum_{i=1}^{M} Y_i^2}{\sqrt{\left(M \times \sum_{i=1}^{M} (X_i)^2 - \left(\sum_{i=1}^{M} X_i\right)^2\right) \times \left(N \times \sum_{i=1}^{M} (Y_i)^2 - \left(\sum_{i=1}^{M} Y_i\right)^2\right)}} \tag{2.40}$$

Où X, Y sont les valeurs des niveaux de gris de deux pixels adjacents, et M le nombre de pairs de pixels, et C_c designe la valeur du coefficient de corrélation suivant une direction.
La figure 2.24 présente un exemple de distribution de pixels dans l'image Lena (512×512), où les pixels sont fortement corrélés selon la direction horizontale.

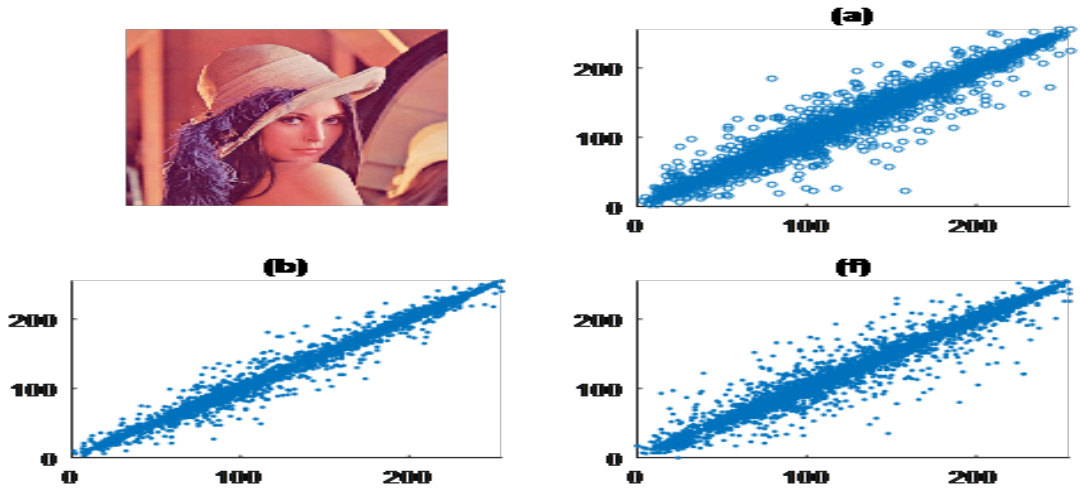


FIGURE 2.24 – Corrélation de l'image Lena dans la direction horizontale

2.7.1.3 Entropie de l'information

L'entropie mesure les informations moyennes par bit dans une image et renseigne sur la quantité d'ordre dans cette dernière. Elle est définie par la relation suivante :

$$S(x) = -\sum_{i=0}^{255} P(x_i)\log_2 P(x_i) \tag{2.41}$$

Où $S(x)$ représente l'entropie de la source du message (x), $P(x_i)$ indique la probabilité d'occurrence de x_i. Pour une image à 256 niveaux de gris, les pixels ont 2^8 valeurs possibles ; l'entropie d'une distribution de valeurs uniforme (cas de l'image cryptée) devrait être égale à 8 de manière idéale [23].

2.7.1.4 Rapport signal sur bruit (Peak Signal-to-Noise Ratio : PSNR)

Le $PSNR$ est un outil utilisé pour quantifier la distorsion entre un signal original x et ce même signal noyé dans un signal bruité y. Dans le cadre du cryptage d'images, cette mesure est basée essentiellement sur le calcul de la différence (mesures de distances) entre l'image

originale et l'image cryptée. Cette mesure renseigne sur le degré de dégradation de l'image originale provoqué par l'application d'une telle méthode de cryptage. Le PSNR est défini mathématiquement par la relation (2.42) [92].

$$PSNR = 10\log_{10}\left(\frac{255^2}{MSE}\right) \quad (dB) \tag{2.42}$$

Où le MSE (Mean Square Error) est l'erreur quadratique moyenne entre l'image cryptée E et l'image originale correspondante I.
La valeur normalisée du MSE notée $NMSE$ est définie par l'équation (2.43) suivante :

$$NMSE = \frac{\sum_{i=1}^{N}\sum_{j=1}^{M}\left[I_D(i,j) - I_E(i,j)\right]^2}{\sum_{i=1}^{N}\sum_{j=1}^{M}\left[I_E(i,j)\right]^2} \tag{2.43}$$

Où $I_D(i,j)$ et $I_E(i,j)$ représentent respectivement les valeurs des pixels (i,j) de l'image décryptée et cryptée.

2.7.2 Analyse différentielle

Un bon crypto-système doit être sensible aux moindres changements opérés sur l'image originale, sinon, il peut être vulnérable aux attaques dites différentielles. La sensibilité d'un algorithme est évaluée à l'aide de deux métriques principales :

- L'UACI (en anglais, *Unified Average Change Intensity*) : il mesure la moyenne de différence d'intensité entre les deux images ;
- Le NPCR (*Number of Pixel Change Rate*) : il mesure le pourcentage du nombre de pixels différents par rapport au nombre total de pixels entre deux images.

Ces deux grandeurs dont définies par les relations 2.44 et 2.45

$$NPCR = \frac{\sum_{i=1}^{m}\sum_{j=1}^{n} D(i,j)}{m \times n} \times 100\,\% \tag{2.44}$$

Avec $D(i,j) = \begin{cases} 0 & si \quad C(i,j) = C'(i,j) \\ 1 & si \quad C(i,j) \neq C'(i,j) \end{cases}$

Où m et n représentent respectivement les dimensions de l'image ; $D(i,j)$ indique la différence entre les pixels correspondants de l'image originale cryptée $C(i,j)$ et l'image originale modifiée

cryptée $C'(i,j)$. La plage de valeur du NPCR d'une image cryptée doit être proche de 100.

$$UACI = \frac{\sum_{i=1}^{m}\sum_{j=1}^{n}|C(i,j) - C'(i,j)|}{255 \times m \times n} \times 100\% \tag{2.45}$$

2.7.3 Analyse de la clé

2.7.3.1 Analyse de l'espace de clé

Les clés de sécurité constituent l'élément central de tout algorithme de chiffrement, car la robustesse de l'algorithme en dépend. La clé d'un algorithme de chiffrement doit être large pour que le crypto-système puisse résister aux attaques exhaustives. Selon Fu C. et al. [94], une clé de taille 10^{30} est suffisante pour résister à une attaque de texte à force brute.

2.7.3.2 Analyse de la sensibilité de la clé

Un algorithme de chiffrement doit être sensible au moindre changement de bit effectué sur la clé afin de résister aux principales attaques. La sensibilité de la clé peut se traduire de la manière suivante :

* Si deux clés légèrement différentes sont utilisées pour crypter la même image, alors les deux images cryptées obtenues doivent être complètement indépendantes l'une par rapport à l'autre ;
* Si l'on modifie la clé de cryptage d'une manière légère, le décryptage de l'image ne peut pas se faire de manière correcte.

2.7.4 Temps d'exécution

Le temps d'exécution est important pour l'évaluation d'un algorithme de cryptage : plus ce temps est court, plus l'algorithme est difficile à le révéler. Le calcul du temps dépendra des caractéristiques du processeur avec lequel les simulations sont faites ainsi que de la complexité de l'algorithme de chiffrement. Pour les crypto-systèmes appropriés aux applications en temps réel, leur temps d'exécution des phases de cryptage et décryptage devrait être de l'ordre de la seconde ou moins, pour de faibles quantités de données. Pour les applications dans les domaines militaire, la télémédecine, systèmes biométriques, les télécommunications par exemple, le temps d'exécution se veut de plus en plus réduit. Dans la pratique, il est nécessaire de comparer le temps d'exécution d'un algorithme avec d'autres développés dans le même environnement (logiciel, performance du processeur, etc.).

2.7.5 Analyse de la complexité de l'algorithme

L'évaluation de la complexité temporelle d'un algorithme de chiffrement permet d'apprécier la performance de ce dernier quant à sa vitesse de chiffrement /déchiffrement. Cette complexité dépend d'une part du nombre d'opérations requises lors du processus de chiffrement. D'autres part, elle est également fonction des performances du PC utilisé lors des simulations, des caractéristiques du processeur, du système d'exploitation, du langage de programmation utilisé, etc. un cryptosystème est d'autant plus intéressant lorsqu'il présente une complexité temporelle faible.

2.7.6 La cryptanalyse

Après la conception d'un nouvel algorithme de cryptage, il est nécessaire d'apprécier sa performance quant aux principales attaques menées par les intrus. Généralement, sans possession de la clé de chiffrement, les intrus utilisent plusieurs procédés pour retrouver l'information cryptée. Entre autres, les attaques les plus opérées sont l'attaque à texte clair choisi (en anglais, Chosen Plain-Text Attack) et l'attaque à texte chiffré choisi (Chosen Cipher-Text Attack).

2.8 Description de l'environnement d'implémentation de l'algorithme proposé dans la contribution 1

Dans cette section, nous présentons la carte intégrée utilisée pour effectuer un début d'implémentation du premier algorithme ainsi que les autres outils associés à la réalisation de cette opération.

2.8.1 Présentation de la carte intégrée STM 32

L'échange des données multimédia d'un point à l'autre (images, audio, vidéo) s'accroit de plus en plus de nos jours et exige un niveau de sécurité adéquat. En réponse à ce besoin, plusieurs auteurs ([95]–[100]) ont développé des cryptosystèmes adaptés aux systèmes embarqués en utilisant des microcontrôleurs, cartes FPGA. De tels dispositifs de sécurité sont assez sollicités dans les domaines de la sécurité, télémédecine, etc.
Dans le cadre de ce travail, nous abordons les possibilités d'implémentation du premier algorithme proposé sur une carte STM32F407ZET6. Telle que la carte Arduino, la carte intégrée STM32F407ZET6 est une carte à microprocesseur disposant d'une interface de carte mémoire SD, d'un port série USB permettant sa connexion avec un ordinateur et plusieurs autres dispositifs externes. Elle peut de ce fait être liée à plusieurs pré-actionneurs, actionneurs ou charges tels que : des lampes /LEDs, moteurs, modules LCD, Wifi et bien d'autres. Lorsque cette carte

est connectée aux modules externes d'affichage, de réception et transmission d'informations, elle est adaptée pour réaliser l"implémentation d'un algorithme de cryptage d'images pour la transmission d'un point à l'autre. Cette carte présente plusieurs atouts, entre autres :

(*i*) Elle est dédiée pour des applications embarquées et systèmes intelligents nécessitant une grande performance ;

(*ii*) Son prix est accessible et elle consomme moins d'énergie ;

(*iii*) C'est un microcontrôleur de 32 bits, Core ARM-M. kernel ;

(*iv*) La puissance de son processeur est supérieure à celle d'une carte FPGA

(*v*) Son environnement de développement (émulateur) dispose d'un ensemble de librairies qui rend sa programmation aisée ;

(*vi*) Son environnement de développement (émulateur) dispose d'un ensemble de librairies qui rend sa programmation aisée ;

Les caractéristiques techniques de la carte STM32F4 sont illustrées dans le tableau 2.2 suivant :

TABLE 2.2 – Caractéristiques techniques du STM32F4

Caractéristique	**Description**	**Caractéristique**	**Description**
Fabricant	STMicroelectronics	Wifi	Oui
$N°$ de Série :	STM32F4	Bluetooth	Non
Core du processeur :	ARM® Cortex®-M3	Tension : d'alimentation	(Vcc/Vdd) 2V $\sim$ 3.6V
Architecture :	32 bits	Périphériques :	DMA, PDR, POR, PVD, PWM, WDT, temporisation
Fréquence :	72 MHz	Convertisseur de données :	A/D 10×12
Nombre de pins I/O :	37	Oscillateur :	Interne
RAM :	20KB	Température fonctionnelle :	$-40°C \sim 85°C$ (TA)
Mémoire FLASH :	64 KB	Bus :	SPI, I2C, UART, CAN,
ADC pins	10		

2.8.2 Matériel utilisé pour un processus d'implémentation de l'algorithme

La phase d'implémentation matérielle de l'algorithme fait appel à des composants matériels ainsi que des éléments logiciels/codes.

- **Composants matériels** : carte STM32F4 (01) ; afficheur LCD de référence 2.8 TFT (01) ; micro carte mémoire SD (01) ; module Wifi de référence Eps8266 (01) ;

- **Eléments Logiciels/codes** : codes de cryptage et décryptage d'images, code de lecture d'image, pilotes de la carte mémoire SD, des boutons poussoirs liées aux clés, des LEDs et de l' afficheur LCD.

La liaison entre la carte STM32F407ZET6 et les périphériques associés se fait à l'aide de différentes interfaces :

— L'interface SDIO relie la carte STM32F407ZET6 à la micro carte mémoire SD ;
— L'interface GPIO permet la connexion de la carte STM32F407ZET6 avec les boutons autorisant le cryptage et le décryptage d'images ;
— L'interface FMSC permet la liaison de la carte STM32F407ZET6 avec l'afficheur LCD.

Le schéma synoptique d'implémentation est illustré à la figure 2.25 suivante :

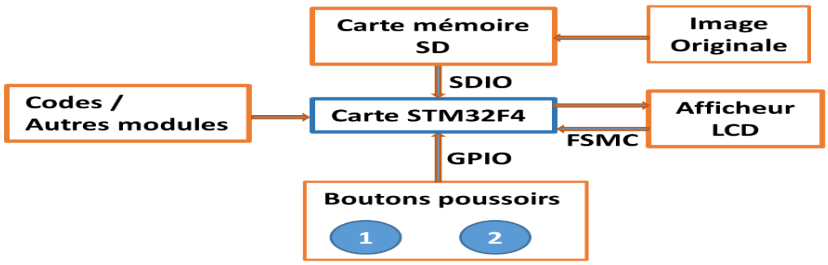


FIGURE 2.25 – Schéma synoptique d'implémentation de l'algorithme

L'image cryptée et affichée au niveau de l'écran LCD peut être transmise à un destinataire : dans ce cas, le schéma synoptique d'implémentation peut être associé à un module Wifi Eps8266 connecté à la carte STM32F407ZET6 via l'interface USART.

La structure matérielle de la carte STM32F407ZET6 et son module Wifi Eps8266 est illustrée à la figure 2.26 suivante :

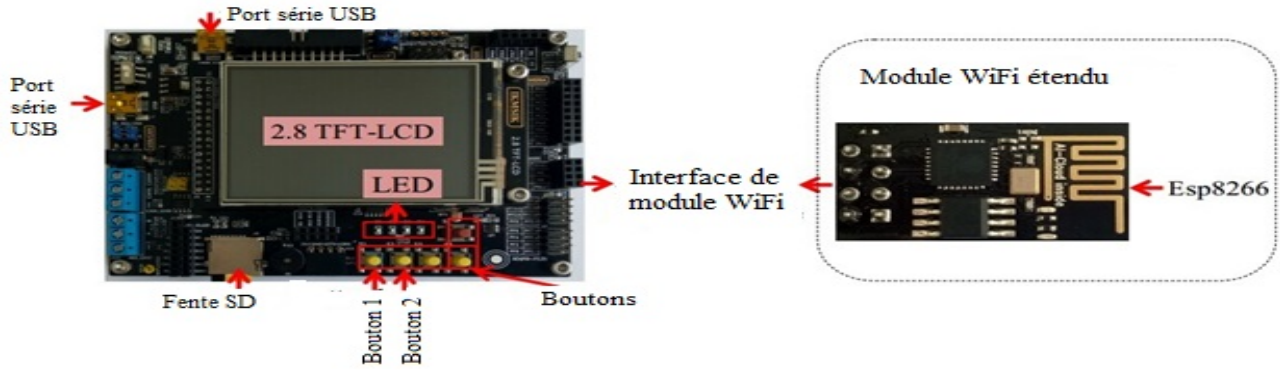


FIGURE 2.26 – Structure matérielle de la carte STM32F4 et du module Wifi Eps8266 [95].

2.8.3 Algorithme d'implémentation sur la carte STM32F407ZET6

L'implémentation du premier cryptosystème proposé sur la carte STM32F407ZET6 peut être effectué selon l'algorithme suivant :

Algorihme d'implémentation sur carte STM32F407ZET6

Entrée : image originale, paramètres de la clé de chiffrement

Sortie : les images cryptée et décryptée affichées sur l'écran LCD

// appuyer la commande $key\_check()$ pour obtenir les paramètres de la clé

1. Void $key\_check(void)$
2. {réinitialiser la clé ;
3. Si (le bouton de la clé est enfoncé)

4 retourner ; // La condition est vraie, elle indique que l'image est en cours
// de traitement et que le programme n'exécutera pas le processus
// de balayage ultérieur. L'idée est d'éviter les multiples traitements
// de la même image causé par une clé répétée.

5. Pour $(i = 0, i \leq Key\_Num\_Count; i++)$
6. {Utiliser la commande $key\_scan(...)$ pour scanner la clé ;
7. Si (le bouton de la clé est appuyé pendant un temps
{Faire appel au gestionnaire de la clé et attribuer une valeur à la clé fin}}
// Utiliser la commande $int\_main(...)$ pour effectuer les processus de cryptage et décryptage de l'image et afficher les images sur l'écran LCD
8. $int\_main(void)$
9. {Effacez le code du système et appelez le code de traitement de l'image ;
10. Utiliser la commande $data\_formatting(...)$ pour obtenir le numéro de version du système ;
11. Utiliser la commande $init\_bsp(...)$ pour initialiser le niveau de la carte, c'est-à-dire les interfaces d'initialisation ;
12. Utiliser la commande $ili9341\_GramScan(...)$ pour mettre en marche l'afficheur LCD ;
13. Utiliser la commande $ili9341\_Clear(...)$ pour définir la couleur de fond de l'écran LCD ;
14. Utiliser la commande $fatfs\_func\_init(...)$ pour initialiser les fonctions de la bibliothèque du système de fichiers ;
15. Utiliser la commande $piclib_init(...)$ pour initialiser les fonctions de la bibliothèque du système d'images ;

16. Si (la carte SD est vide)
17. {Allumer toutes les LED et afficher ''erreur de carte SD '' sur l'écran LCD ;}
18. Utiliser la commande $list\_files(...)$ pour trouver les fichiers images en SD et mettre leurs noms dans la mémoire cache ;
19. Utilisez les commandes $comClearTxFifo(...)$ et $comClearRxFifo(...)$ pour effacer les données de la mémoire cache des ports série ;
20. Fixez la valeur des variables d'état à 0, ainsi que celles des paramètres $client\_connect$, key_value, pic_handl, et pic_entry ;
21. Utilisez la commande $timer\_start\_auto(...)$ pour mettre en marche les temporisateurs numérotées par 0 et 1, puis définir l'intervalle de temps à s'écouler entre ces temporisateurs ;
22. **Quand** (1)
23. {**si** (le temporisateur numéroté 0 est mis en marche)
24. {Faire clignoter de manière continue la LED $N°4$
25. {**si** (le temporisateur numéroté 1 est mis en marche)
26. {**Si** ($pic\_handl == 0$) {effectuer $key\_check(...)$}}
27 **Entrer** (valeur de la clé)

// phase de cryptage

28. {1^{er} cas : Utiliser la commande $sprintf(...)$ pour lire le nom de fichier de l'image originale dans la zone de la mémoire cache
29. Faire $pic\_handl = 1$;
30. Utiliser la commande $ili9341\_Clear(...)$ pour effacer toutes les données sur l'écran LCD et laisser $pic\_entry = 1$;
31. Utiliser la commande $ai\_load\_picfile(...)$ pour lire l'image originale ;
32. Utiliser les commandes $JPEG\_decode(...)$, $jd\_decomp(...)$, $jd\_prepare(...)$, et $f\_open(...)$ pour décoder l'image ;
33. Utiliser le code $pic\_encrypt\_algorithme1(...)$ pour crypter l'image originale ;
34. Utiliser la commande $ili9341\_GetPointPixel(...)$ pour afficher l'image cryptée sur l'écran LCD et laisser $pic\_entry = 0$;
35. Utiliser la commande $get\_pic\_info(...)$ pour obtenir les données cryptées et la commande $sprinft(...)$ est utilisée pour les nommer ;
36. Utiliser les codes $JPG\_encode(...)$, $jpeg\_creat\_compress(...)$ pour décoder les données cryptées et enregistrer l'image cryptée dans la carte SD ;
37. $pic\_handl = 0$ et allumer la LED $N°1$ de manière continue ;
// le 1^{er} cas est terminé.

// phase de décryptage

38. {2^{e} cas : Utiliser la commande $sprintf(...)$ pour lire le nom de fichier de l'image cryptée ;

39. Faire (29) et (30) ;
40. Utiliser les commandes $JPEG\_decode(...)$, $jd\_decomp(...)$, $jd\_prepare(...)$, et $f\_open(...)$ pour décoder l'image cryptée ;
41. Utiliser le code $pic\_decrypt\_algorithme1(...)$ pour crypter l'image originale ;
42. Utiliser la commande $ili9341\_GetPointPixel(...)$ pour afficher l'image cryptée sur l'écran LCD et laisser $pic\_entry = 0$;
43. Utiliser les codes $JPG\_encode(...)$, $jpeg\_creat\_compress(...)$ pour décoder les données décryptées et enregistrer l'image décryptée dans la carte SD ;
44. $pic\_handl = 0$ et allumer la LED $N°2$ de manière continue ;

// l^e, 2^e cas est terminé.

fin ;}}}

Fin algorithme

Cet algorithme d'implémentation peut être simplifier en adoptant l'organigramme suivant :

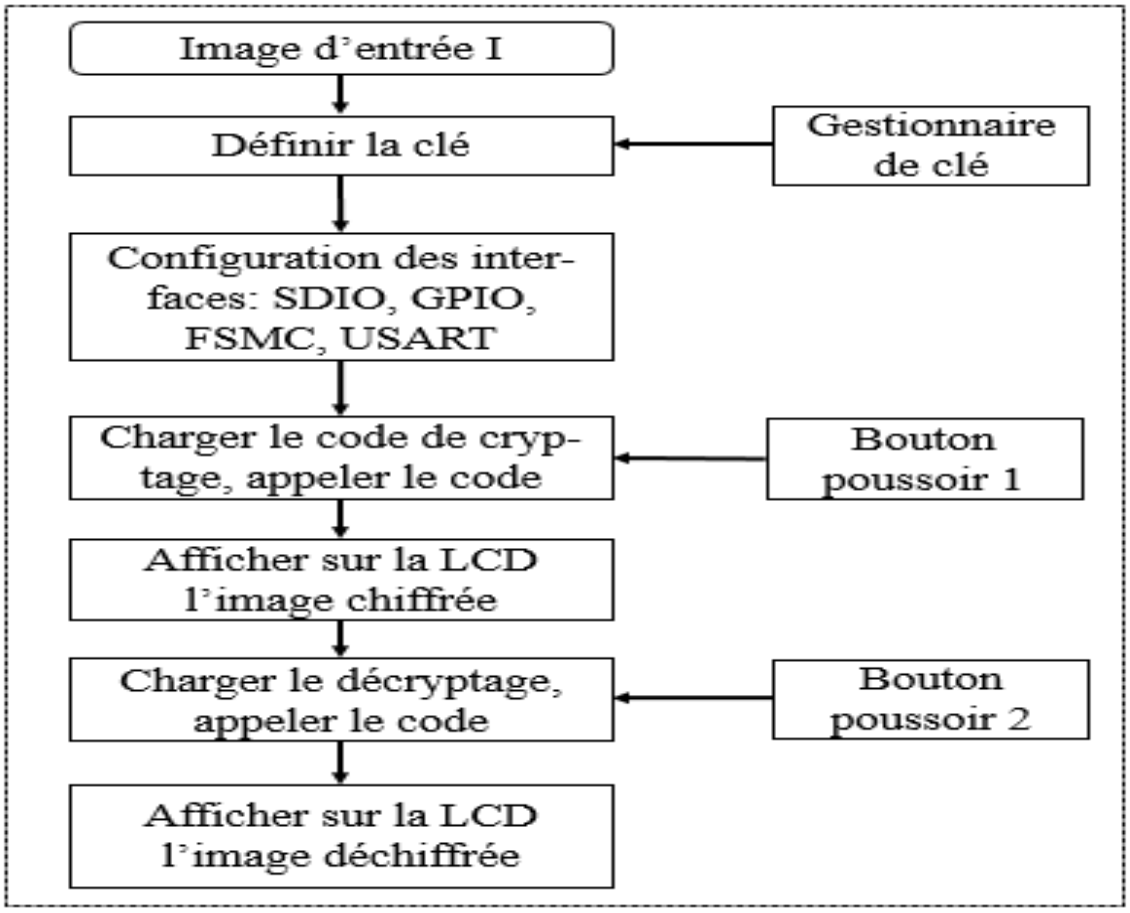


Schéma bloc d'implémentation

Conclusion

Le présent chapitre avait pour objectif, d'une part, de présenter les différents outils que nous avons utilisés pour développer les deux algorithmes proposés basés sur la fusion d'images, et d'autres part, de décrire les algorithmes proposés avec les métriques d'évaluation associées et de présenter l'environnement de développement pour l'implémentation d'un des cryptosystèmes proposé sur une carte intégrée. Tout au long de notre développement, nous avons premièrement fait une présentation de l'organisation des crypto-systèmes basés sur la fusion d'images et analysé quelques algorithmes récents utilisant la fusion dans le processus de chiffrement. Ensuite, nous avons présenté nos contributions portant sur deux algorithmes de cryptage basés sur le mixage ou la fusion d'images, tous bâtis selon la structure permutation-diffusion. Nous notons que ces algorithmes ont l'atout de chiffrer plusieurs images simultanément, de manière robuste sans dégrader de manière significative la qualité d'images reconstruites. Enfin, une présentation des outils et métriques d'analyse de crypto-systèmes a été faite, suivie de la présentation de la carte intégrée STM32F407ZET6 ainsi que de l'algorithme d'implémentation du cryptosystème sur cette carte. De manière précise, les métriques évoquées dans ce chapitre permettront de valider les résultats et simulations obtenus des deux algorithmes proposés. Le chapitre trois qui suit est ainsi consacré aux résultats et analyses.

Chapitre III

Évaluation des algorithmes de chiffrement proposés et analyse expérimentale

Introduction

Dans ce chapitre, nous présenterons les résultats et analyses obtenus en vue d'apprécier les performances des algorithmes proposés. Nous débuterons par la présentation des éléments qui confirment le bon comportement aléatoire des cartes chaotiques utilisées ; ensuite, la présentation de la base de données d'images utilisée sera faite. Cette base contient des images de différentes taille, et principalement les images médicales. Enfin, l'évaluation des algorithmes de chiffrement proposés est faite à l'aide des métriques évoquées au chapitre deux, portant sur les tests et analyses statistique, différentielle et cryptanalyse.

3.1 Caractère pseudo-aléatoire des cartes de données utilisées : Cartes Logistique-May, Tent-May, May-Gaussian et Gaussian-Gompertz

Ces nouvelles cartes ont été construites à base des cartes 1D Logistique, May, Tent, Gompertz et Gaussienne, afin de combler les limitations de ces dernières lorsqu'elles sont utilisées seules dans les crypto-systèmes. Les détails concernant ces cartes peuvent être consultées dans les travaux de Y. Kamdeu et al.[90].
Les outils utilisés pour attester du caractère pseudo-aléatoire de ces générateurs de nombres sont l'exposant de Lyapunov et le diagramme des bifurcations. Le tableau 3.1 résume les traits particuliers de ces cartes.

TABLE 3.1 – Caractéristiques des cartes Logistique-May, Tent-May, May-Gaussian et Gaussian-Gompertz

Cartes	**Exposant de Lyapunov**	**Aspect de la courbe de bifurcation**	**Plage de valeur du paramètre de contrôle**
Logistique-May	8.3	Gamme chaotique large et uniforme	$r \in [0, 5]$
May-Gaussian	[2.5,5.6]	Gamme chaotique large et uniforme	$r \in [0, 5]$
Gaussian-Gompertz	2.5	Gamme chaotique large et uniforme	$r \in [0, 5]$

Selon le tableau 3.1, les nouvelles cartes combinées présentent des atouts, notamment, leur exposant de Lyapunov est plus grand (tableau 3.2), leur diagramme de bifurcation présente une large plage chaotique avec une distribution uniforme de valeurs. Ces considérations précédentes réduisent l'effet transitoire des séries de valeurs issues de ces cartes, et font d'eux de meilleurs générateurs de nombres Pseudo-aléatoires, comparés à leurs semences.

TABLE 3.2 – Exposant de Lyapunov des cartes sources 1D et des cartes combinées [90].

Cartes sources	**Exposant de Lyapunov**	**Cartes combinées**	**Exposant de Lyapunov**
Logistique-	0.6	Logistique-May	8.3
May	0.4	May-Gaussian	≥ 2.5
Gaussienne	0.5	Gaussienne-Gompertz	2.5
Gompertz	0.7		

Bien plus, pour confirmer le bon caractère pseudo-aléatoires des données issues de ces cartes, les tests du NIST (National Institute of Standards and Technology) ont été menés tels qu'indiqués dans le tableau 3.3. L'on peut y observer que la série des douze tests effectués a conduit au succès de ces derniers, révélant ainsi le bon caractère pseudo-aléatoire des cartes de données.

TABLE 3.3 – Résultat des tests de NIST SP800-22 en mode SC. Le symbole √ désigne le terme succès, LM : carte Logistique-May, LSM : carte Logistique-Sine, MG : carte May-Gompertz, GG : carte Gaussienne-Gompertz.

Mode					**SC**			
Test	p-valeur	Res	p-valeur	Res	p-valeur	Res	p-valeur	Res
Test		**LSM**		**LM**		**MG**		**GG**
Test d'entropie approximative	0.1203	√	0.45393	√	0.0907	√	0.41560	√
Test de fréquence par bloc	0.3601	√	0.6929	√	0.9972	√	0.678415	√
Test de somme cumulative	0.3433	√	0.78621	√	0.5263	√	0.9014	√
Transformé de Fourier rapide	0.7263	√	0.87531	√	0.1554	√	0.82670	√
Test de fréquence	0.1665	√	0.98147	√	0.0409	√	0.99438	√
Exécution aléatoire	0.9195	√	0.195257	√	0.9896	√	0.77856	√
Exécution aléatoire variable	0.7801	√	0.14358	√	0.7604	√	0.25167	√
Test de longue série de 1	0.6126	√	0.98654	√	0.2417	√	0.97729	√
Test de rang	0.3311	√	0.54702	√	0.0528	√	0.5873	√
Test de complexité	0.7541	√	0.08945	√	0.1551	√	0.87246	√
Test de série	0.5467	√	0.42962	√	0.5869	√	0.41757	√
Test statistique universel	0.2471	√	0.38277	√	0.9792	√	0.37051	√

Les autres utilisées dans ces travaux, à savoir la carte de Henon et la carte Linéaire chaotique par Morceaux (PWLCM) révèlent de par leur attracteur et diagramme de bifurcation, une distribution uniforme de valeurs, presque répartie sur toute la plage du paramètre de contrôle. Cela montre à suffisance leur bon caractère de générateur de nombres pseudo-aléatoires.

3.1.1 Présentation de la base de données d'images

Les principales images sur lesquelles les tests sont effectués sont les images médicales de différentes tailles. Toutefois, afin de confirmer les résultats, nous appliquons d'abord les tests sur les images standards de la littérature très utilisées en cryptographie (images à niveau de

gris et images couleur) telles que : Lena, Mandrill, Cameraman, Pepper, Plane, Airport, etc. Les différentes images utilisées dans les deux algorithmes sont illustrées ci-dessous par les figures 3.1 et 3.2.

- **Images à niveau de gris et images couleurs**

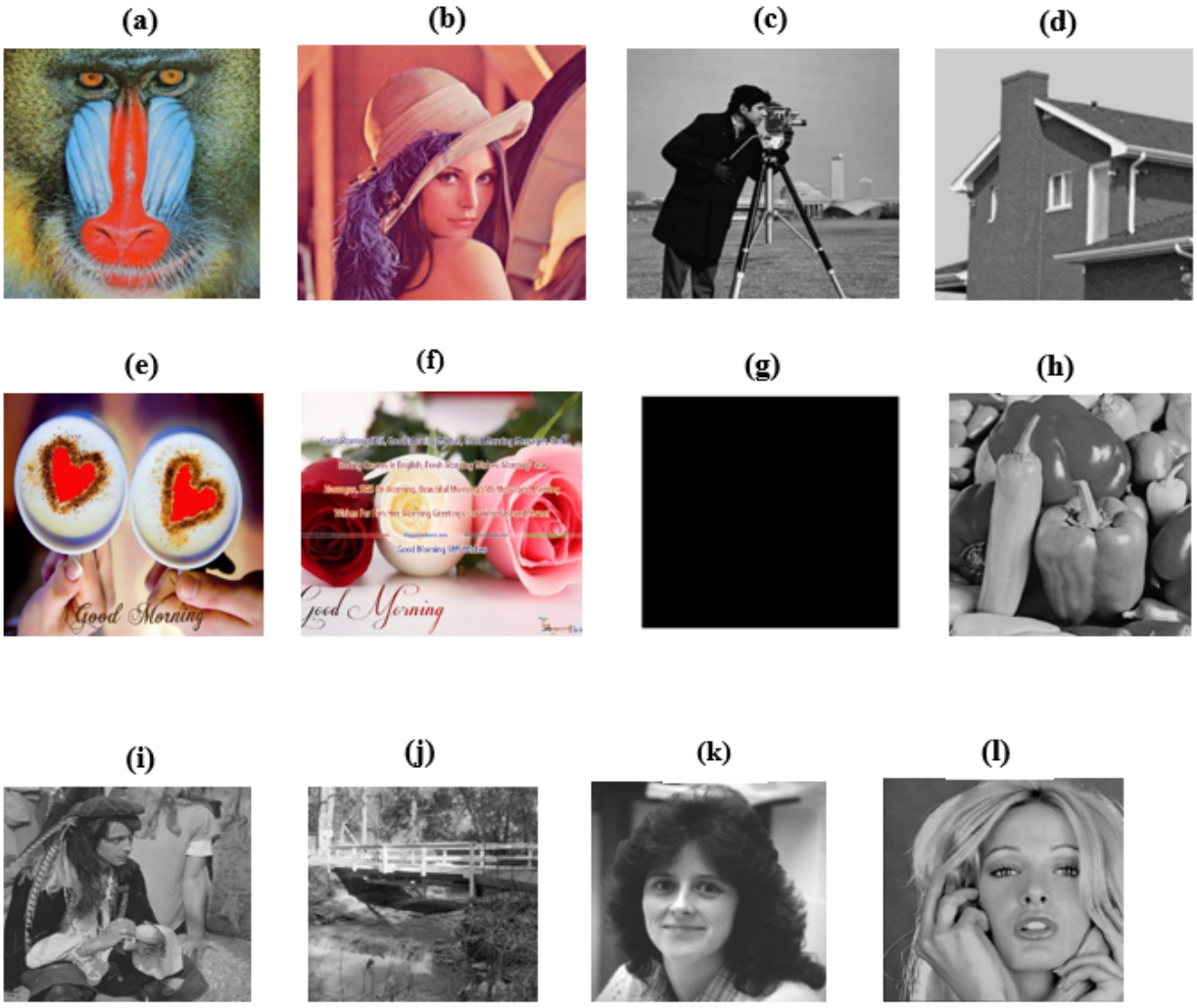


FIGURE 3.1 – Images à niveaux de gris, et images couleurs.

Légende : (a) Mandrill (couleur, 512×512), (b) Lena (couleur, 512×512), (c) Cameraman (gray, 256×256), (d) house (gray, 512×512), (e) Heart (couleur, 640×480), (f) Bloom (couleur, 640×480), (g) Black image (binaire, 512×512) ; (h) Pepper (gray, 512×512) , (i) man (gray, 512×512), (j) Bridge (gray, 512×512), (k) Woman (gray, 512×512), (l) Girl ((gray, 512×512).

- **Images à niveau de gris et images couleurs**

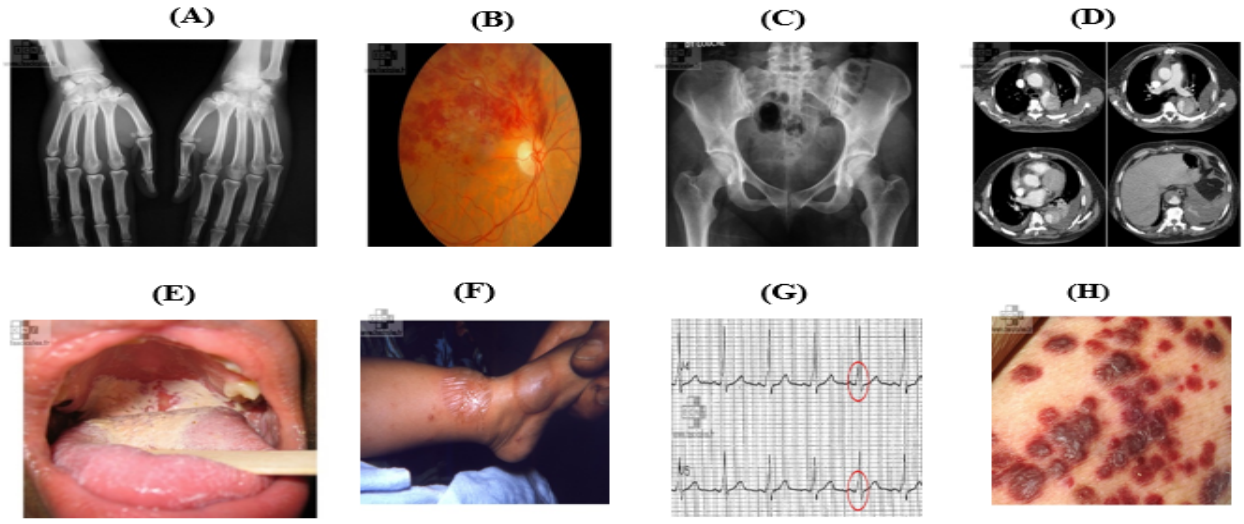


FIGURE 3.2 – Images médicales.

Légende : (A) Phalanges (880×660) , (B) œil (900×580) , (C) Bassin (880×660), (D) Thorax (880×660), (E) Langue (880×660) , (F) Pied (880×660), (G) ECG (880×390), (H) peau (880×660).

3.2 Analyse de sécurité de la contribution 1 : " Transmission sécurisée d'images médicales pour la télémédecine"

Les tests sont menés sur différentes images de la base de données, de tailles $n \times m$ avec $n = m$ pour certains, et $n \neq m$, pour d'autres afin d'explorer les différents cas.
Ces tests et analyses de sécurité ont été effectués avec un ordinateur ayant un processeur Core (TM) i7-353U, 2.5 GHz de fréquence, sur la plateforme MATLAB 2016b. Les conditions initiales pour chiffrer la première image ont été calculées à partir du ''haché" de l'image Lena (512×512).
SHA 256 (Lena) : c9ca3d339e38fd573590cda14ae30c45c7d89f44703e42b8ba2ab04c5bb0557f
Les résultats ont été analysés en termes d'analyses statistiques, d'attaque à force brute, attaque à texte clair, attaque différentielle, attaque à texte choisi, temps d'éxécution et sensibilité au bruit. Les valeurs des differents paramètres de la clé sont regroupés dans le tableau 3.4 suivant :

TABLE 3.4 – Paramètres de la clé

Paramètres de la clé	
Algorithme 1	$x_{01}^{R} = 0.99295\,;\ x_{01}^{C} = 0.98874\,;\ r_{01}^{R} = 4.99871\,;\ r_{01}^{C} = 4.98601\,;$ $a = 1.39012\,;\ x_{0}^{'} = 0.54289\,;\ y_{0}^{'} = 0.12658$
Algorithme 2	$x_{1}^{'} = 0.97402\,;\ x_{2}^{'} = 0.96719\,;\ r_{1} = 3.97238\,;\ r_{2} = 3.95973$
Fusion	$x_{01}^{''} = 0.88954\,;r_{1}^{''} = 4.99961\,;x_{02}^{''} = 0.91058\,;\ r_{2}^{''} = 3.92751$

Les images chiffrées deux à deux sont présentées sur les figures 3.3 et 3.4. Les deux images doivent être de même taille, et vont produire deux images hybrides, dont l'une comporte les informaions de l'autre.

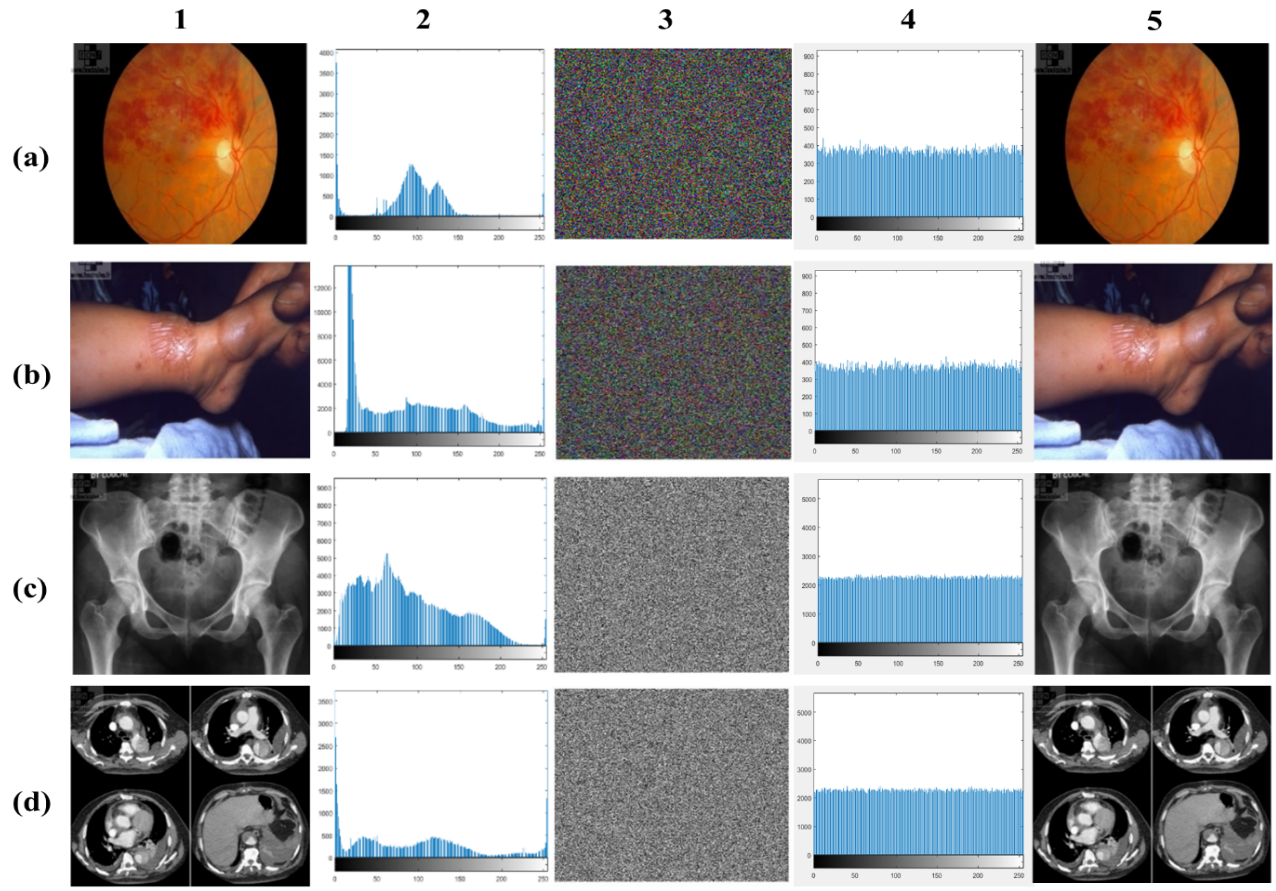


FIGURE 3.3 – De la gauche vers la droite dans le sens de la colonne (1 à 5) : Images médicales d'entrée et leurs histogrammes respectifs, images chiffrées et leurs histogrammes, images décryptées. a-1 Eye.tiff (900×900) , b-1 Leg.tiff (900×900), c-1 Pelvis.jpg (880×660) , d-1 Thorax.jpg (880×660),a-3 image hybride 1 cryptée (Eye-Leg), b-3 image hybride 2 cryptée (Eye-Leg), c-3 image hybride 1 cryptée (Pelvis-Thorax), d-3 image hybride 2 cryptée (Pelvis-Thorax), a-5 à d-5 images décryptées.

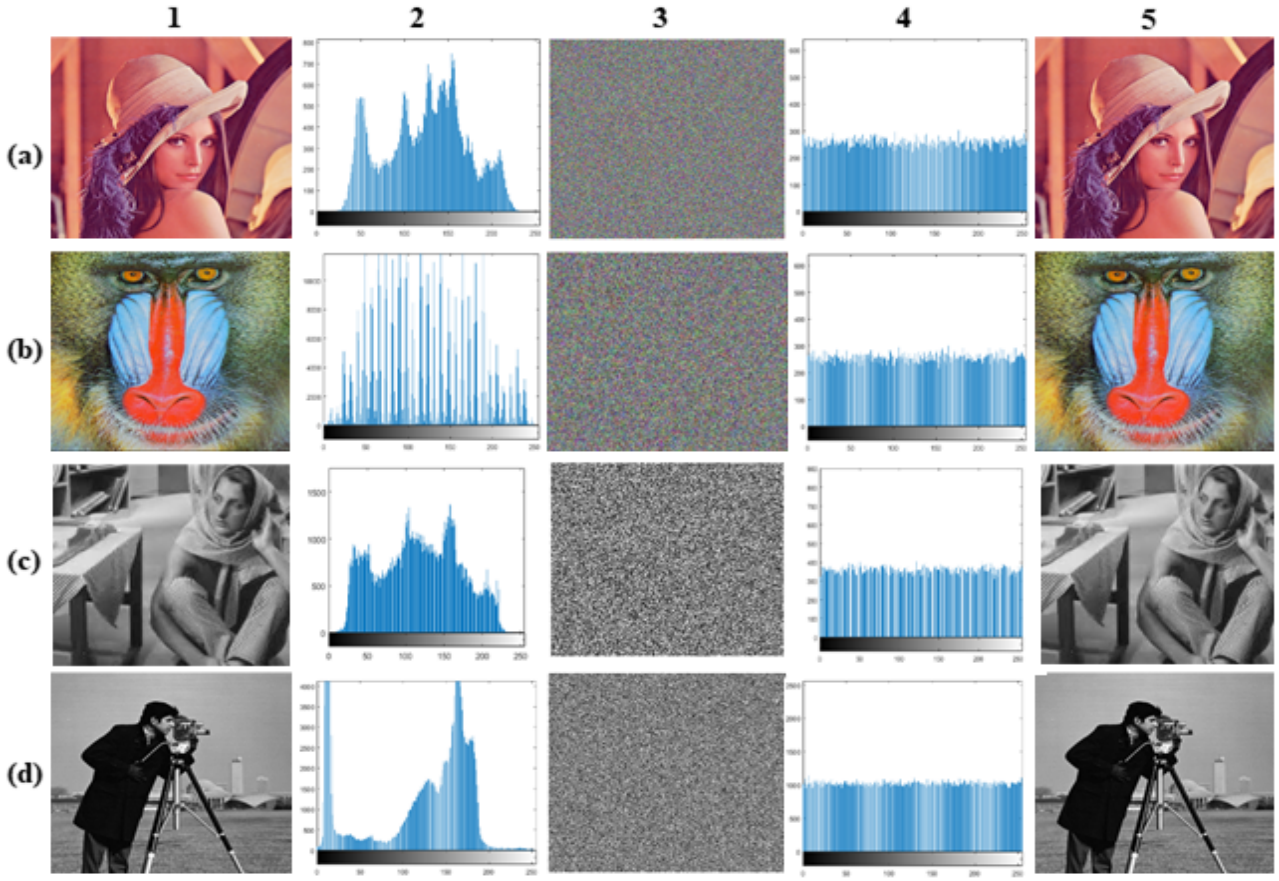


FIGURE 3.4 – De la gauche vers la droite dans le sens des colonnes (1 à 5) : Images d'entrée standards et leurs histogrammes ; images cryptées et leurs histogrammes ; images décryptées. a-1 Lena.tiff (512 × 512) ; b-1 Baboon.tiff (512 × 512) ; c-1 Barbara.png (256 × 256) ; d-1 Cameraman.png (256 × 256) ; a-3 image hybride 1 cryptée (Lena-Baboon) ; b-3 image hybride 2 cryptée (Lena-Baboon) ; c-3 image hybride 1 cryptée (Black-Cameraman) ; d-3 image hybride 2 cryptée (Black-Cameraman) ; a-5 à d-5 images décryptées.

3.2.1 Analyse statistique

3.2.1.1 Histogramme et variance d'histogramme

Selon le standard, l'histogramme d'une image cryptée doit avoir une distribution uniforme. Les figures 3.3 et 3.4 révèlent pour toutes les images cryptées que cette condition est satisfaite. Afin de confirmer cette assertion, nous avons effectué le calcul de la variance d'histogramme

des images chiffrées avec l'équation 3.1. Le tableau 3.5 présente les valeurs obtenues.

$$Var(z) = \frac{1}{n^2}\sum_{i=1}^{n}\sum_{j=1}^{n}\frac{1}{2}(z_i - z_j)^2 \tag{3.1}$$

Où Z est le vecteur des valeurs d'histogramme, $Z = \{z_1, z_2, ..., z_{256}\}$; z_i et z_j représentent le nombre de valeurs de pixels égaux à i et j respectivement.

TABLE 3.5 – Variance d'histogramme d'images chiffrées

Image	Image originale	Image crypté	cryptosystème proposé	[93]	[90]	[12]
Eye (900×900)	6.321×10^6	Image hybride 1 (Eye-Leg)	3458.1	-	-	-
Leg (900×900)	9.720×10^6	Image hybride 2 (Eye-Leg)	3237.8	-	-	-
Thorax (880×660)	4.826×10^6	Image hybride 1 (Thorax-Pelvis)	4065.2	-	-	-
Pelvis (880×660)	2.863×10^6	Image hybride 2 (Thorax-Pelvis)	4129.3	-	-	-
Lena (512×512)	6.255×10^6	Image hybride 1 (Lena-Baboon)	3214.7	1050.87	5450.87	1077
Baboon (512×512)	6.193×10^6	Image hybride 2 (Lena-Baboon)	3764.2	1058.12	-	971
Barbara (256×256)	6.27×10^6	Image hybride 1 (Barbara-Cameraman)	5236.3	1271.65	-	-
Cameraman (256×256)	5.872×10^6	Image hybride 2 (Barbara-Cameraman)	5180.6	-	5482.61	-

Selon le tableau 3.5, la valeur moyenne de variance d'histogramme d'images chiffrées est de 4600 qui est inférieure à 5000, ce qui constitue une condition satisfaisante pour obtenir un histogramme uniforme des images chiffrées [6]. Nous notons que l'analyse d'histogramme du crypto-système proposé ne présente pas de failles de sécurité.

3.2.1.2 Analyse de corrélation des pixels adjacents

Une image chiffrée par un algorithme robuste devrait présenter une forte décorrélation des pixels dans toutes les directions (Horizontale, Verticale et Diagonale). Les valeurs des coefficients de corrélation des images chiffrées sont mentionnées dans le tableau 3.6. L'on peut observer que ces valeurs sont toutes proches de zéro comme souhaité, traduisant ainsi une faible

corrélation entre les pixels adjacents d'images chiffrées. De plus, ce résultat peut est confirmé en observant la figure 3.5 qui présente la distribution de pixels de l'image Pelvis avant et après chiffrement. Les pixels sont fortement décorrélés après cryptage, d'où l'algorithme proposé est sécurisé contre les attaques par corrélation.

TABLE 3.6 – Coefficient de corrélation d'images chiffrées.

Image	Taille	Test	Image originale	Image cryptée	cryptosystème proposé	[8]	[12]
Eye	(900	CH	0.9715	Image hybride 1	0.0040	-	-
	×900)	CV	0.9984	(Eye-Leg)	0.0012		
		CD	0.9607		-0.0019		
Leg	(900	CH	0.9986	Image hybride 2	0.001	-	-
	×900)	CV	0.9926	(Eye-Leg)	-0.002		
		CD	0.9912		0.0018		
Pelvis	(880	CH	0.9833	Image hybride 1	-0.002	-	-
	×660)	CV	0.9985	(Pelvis-Thorax)	0.0014		
		CD	0.9846		0.017		
Thorax	(880	CH	0.9220	Image hybride 2	0.0022	-	-
	×660)	CV	0.9761	(Pelvis-Thorax)	-0.0014		
		CD	0.9162		0.0009		
Lena	(512	CH	0.9853	Image hybride 1	0.001	0.001	0.00145
	×512)	CV	0.9719	(Lena-Baboon)	-0.002	0.003	0.00237
		CD	0.9590		0.0018	-0.006	0.00022
Baboon	(512	CH	0.9090	Image hybride 2	0.002	0.002	-0.0015
	×512)	CV	0.8989	(Lena-Baboon)	-0.0014	-0.004	-0.0012
		CD	0.8610		0.0017	0.008	0.00035
Camera-	(256	CH	0.9900	Image hybride 1	0.008	-	-
man	×256)	CV	0.9831	(cameraman-Bar-	0.012		
		CD	0.9733	bara)	-0.002		
Barbara	(256	CH	0.9715	Image hybride 2	-0.0015	-	-
	×256)	CV	0.9984	(cameraman-Bar-	0.0013		
		CD	0.9607	bara)	0.0014		

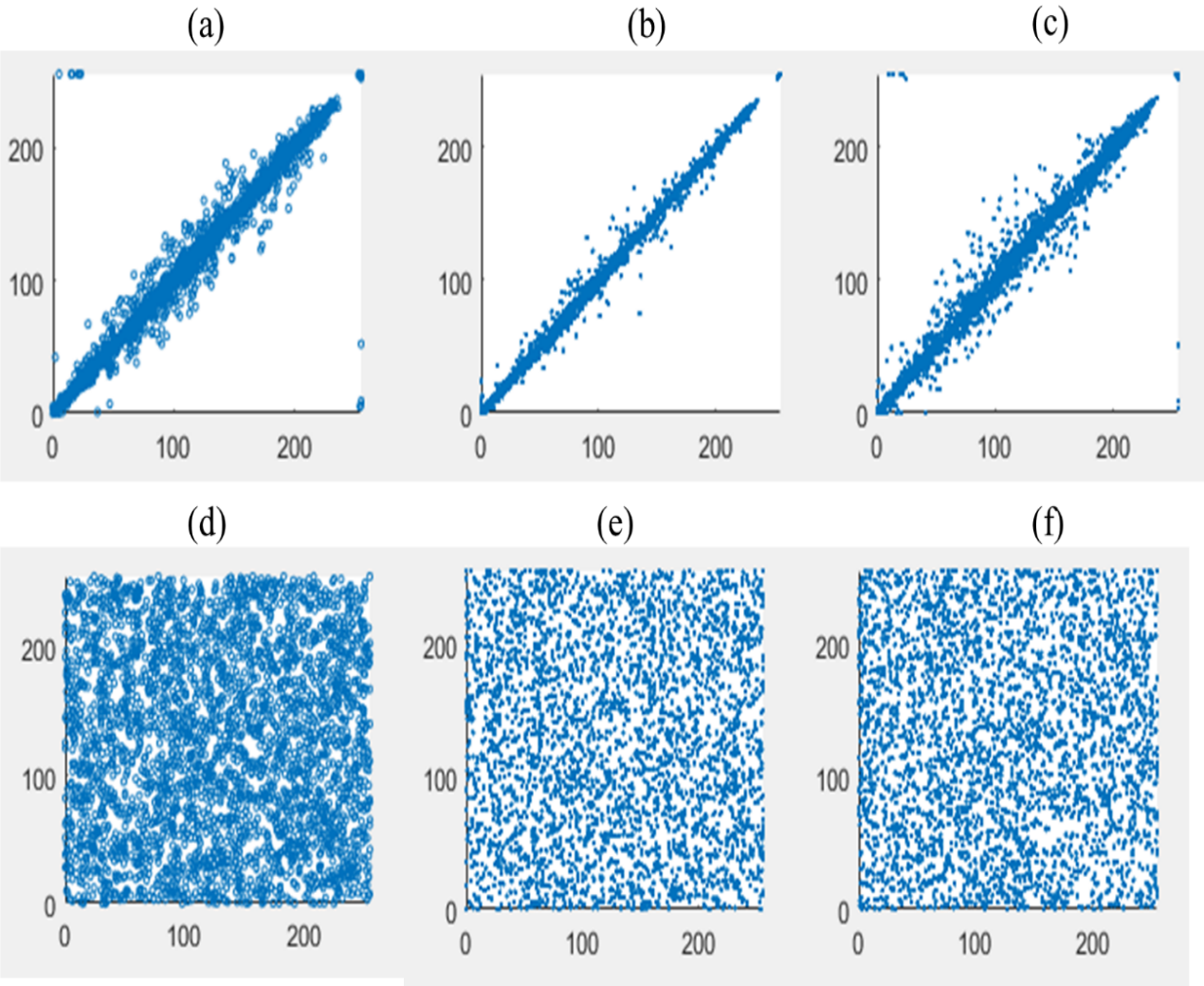


FIGURE 3.5 – Distribution des coefficients de corrélation de l'image Pelvis (a, b, c) Tracé du coefficient de corrélation horizontal, vertical and diagonal corrélation de l'image originale. (d, e, f) Tracé du coefficient de corrélation horizontal, vertical and diagonal de l'image cryptée.

3.2.1.3 Analyse d'entropie de l'information

L'entropie des images chiffrées est déterminée par la relation 2.48. Les résultats sont consignés dans le tableau 3.6. Selon le standard, l'entropie d'une image chiffrée devrait être proche de la valeur 8. En parcourant les valeurs du tableau 3.7 et après comparaison avec quelques-unes de la littérature, les valeurs obtenues sont satisfaisantes, et traduisent le bon degré de désordre dans les images chiffrées. De ce fait, une attaque sur le crypto-système proposé basée sur l'entropie ne peut fonctionner.

TABLE 3.7 – Entropie de quelques images chiffrées

Image	Taille	Image cryptée	Entropie d'images hybrides	[90]	[12]
Eye	(900×900)	Image hybride 1 (Eye-Leg)	7.9993	-	-
Leg	$(900 \times 900$	Image hybride 2 (Eye-Leg)	7.9993	-	-
Pelvis	(880×660)	Image hybride 1 (Pelvis-Thorax)	7.9994	-	-
Thorax	(880×660)	Image hybride 2 (Pelvis-Thorax)	7.9994	-	-
Lena	(512×512)	Image hybride 1 (Lena-Baboon)	7.9993	7.9994	7.996
Baboon	(512×512)	Image hybride 2 (Lena-Baboon)	7.9993	7.9993	7.999
Barbara	(256×256)	Image hybride 1 (Barbara-cameraman)	7.9994	7.9998	-
Cameraman	(256×256)	Image hybride 2 (Barbara-cameraman)	7.9994	7.9971	-

3.2.2 Analyse de la clé

3.2.2.1 Analyse de l'espace de la clé

L'espace de clé est le nombre total de clés qui est utilisé dans un crypto-système. Dans l'algorithme proposé, les paramètres de la clé sont mentionnés dans le tableau 3.1 et comportent 14 éléments. En considérant la précision décimale fixée à 10^{-15} , l'espace de clés de l'algorithme proposé est évalué à $10^{15\times14} = 10^{210}$. Nous référant à Fu C. [101], notre espace clé est supérieur à 10^{30}, donc assez large pour résister aux attaques à force brute.

3.2.2.2 Analyse de la sensibilité de la clé

Afin de résister aux attaques à texte clair choisi ou à image chiffrée choisie, un bon crypto-système doit être sensible au moindre changement de la clé. Afin de vérifier ce test de sensibilité, à partir de la clé originale K_1, nous avons modifié la valeur du paramètre $r_{01}^R = 4.99871$ à $4.99871 + 10^{-15}$ (juste en modifiant un seul bit) en laissant le reste de paramètres inchangés, afin d'obtenir la clé K_2. De la meme manière, K_3 et K_4 sont obtenues respectivement en modifiant le dernier bit de $x_1^{'}$ et $r_2^{''}$.

Les images décryptées avec mauvaises clés K_2, K_3 et K_4 de l'image hybride (Pelvis-Thorax) sont illustrées à la figure 3.6. Le tableau 3.8 présente le pourcentage de différence entre les images chiiffrées avec les trois clés. Ces résultats revèlent que les images chiffrées sont assez différentes de 99.632% en Moyenne, ainsi le cryptosystème présente une forte sensibilité à la clé de chiffrement.

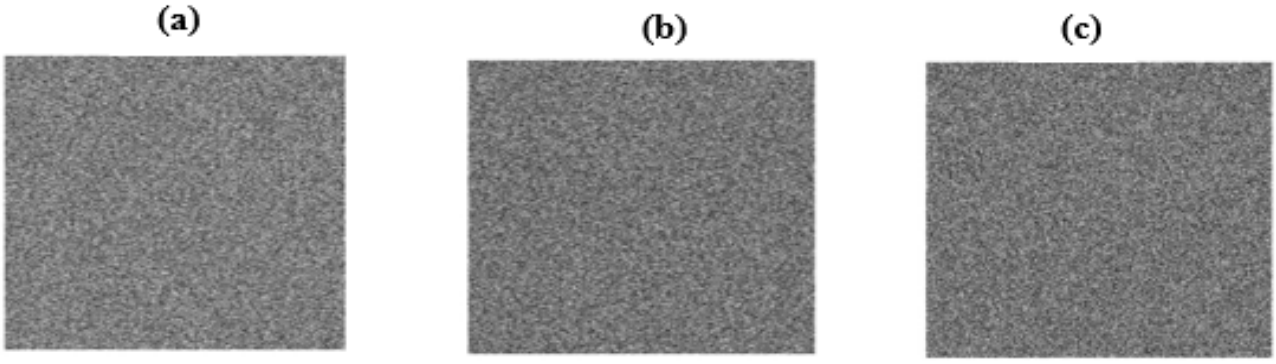


FIGURE 3.6 – Image (Pelvis-Thorax) décryptée avec les mauvaises clés, (a) K_2 , (b) K_3 , (c) K_4.

TABLE 3.8 – Tests de sensibilité de la clé

Clé	Sensibilité de la clé (%)	[90]	[102]
K_1 Vs K_2	99.61	99.61	99.65
K_1 Vs K_3	99.63	99.62	99.60
K_1 Vs K_4	99.58	99.65	99.59

3.2.3 Analyse différentielle

La sensibilité au moindre changement du pixel est évaluée en calculant les valeurs de l'UACI et le NPCR. Les résultats sont relevés dans le tableau 3.9 suivant. Les valeurs moyennes de l'UACI et du NPCR sont respectivement 33.50 et 99.62. Ces valeurs comparées à certaines de la littérature ([8], [90], [93]) sont satisfaisantes. De plus, les valeurs de l'UACI et du NPCR en fonction de différentes valeurs de la clé de chiffrement sont illustrées à la figure 3.7 où l'on peut y observer que les valeurs moyennes de ces deux métriques son très proches de celles évoquées précédemment. Ces résultats prouvent à suffisance que le cryptosystème proposé n'est pas vulnérable contre l'attaque par analyse différentielle.

TABLE 3.9 – Valeurs de NPCR et UACI

Image	UACI(%)	NPCR(%)
Image hybride 1 (Eye-Leg)	33.39	99.59
Image hybride 2 (Eye-Leg)	33.40	99.61
Image hybride 1 (Pelvis-Thorax)	33.47	99.62
Image hybride 2 (Pelvis-Thorax)	33.49	99.63
Image hybride 1 (Lena-Baboon)	33.42	99.62
Image hybride 2 (Lena-Baboon)	33.40	99.61
Image hybride 1 (Cameraman-Barbara)	33.52	99.63
Image hybride 2 (Cameraman-Barbara)	33.35	99.62
Lena [90]	33.46	99.62
Baboon [8]	33.55	99.61
Cameraman [93]	33.72	99.62

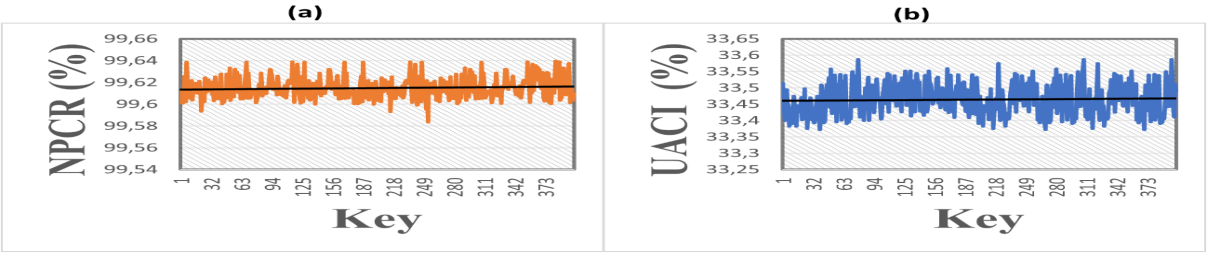


FIGURE 3.7 – Valeurs moyennes du NPCR (a) et de l'UACI (b) en fonction de la clé de chiffrement.

3.2.4 Temps d'exécution

Le temps d'exécution pour chiffrer deux images par le crypto-système proposé est présenté dans le tableau 3.10. La moyenne de temps pour chiffrer simultanément les images Pelvis et Thorax de tailles 880×660 est de 0.7316s. Nous observons que le temps de chiffrement est satisfaisant comparé aux autres algorithmes du même type ([48], [52], [55]) , et est adapté pour les applications en temps réel.

TABLE 3.10 – Temps de cryptage de deux images

	Algorithme proposé				
Images originales	Algorithme 1 Temps (ms)	Algorithme 2 Temps (ms)	Images cryptées	Temps de mixage (ms)	Temps total (Seconde)
Eye (900×900)	321.7	-	Image hybride	223	0.7722
Leg (900×900)	-	227.5	Eye-Leg		
Pelvis (880×660)	308.4	-	Image hybride	210.3	0.7316
Thorax (880×660)	-	212.9	Pelvis-Thorax		
Lena (512×512)	205.6	-	Image hybride	168.7	0.5575
Baboon (512×512)	-	183.2	Lena-Baboon		
Cameraman (256×256)	190.51	-	Image hybride	172.5	0.5506
Barbara (256×256)	-	187.63	Cameraman-Barbara		

3.2.5 Analyse de la complexité

L'évaluation de la complexité spatiale d'un cryptosystème dépend de la façon dont ce dernier a été conçu. Les opérations principales du cryptosystème proposé se résument en deux étapes majeures : à la première étape, les deux images originales d'entrée (de tailles $(M \times N)$ sont chiffrées simultanément, la première image suivant un processus de permutation et diffusion et la seconde selon un processus de diffusion. Finalement à la deuxième phase, les deux images chiffrées précédemment sont mixées à travers une approche de diffusion. En considérant les opérations principales du cryptosystème proposé, il est à noter que la permutation ne nécessite pas d'autres calculs en dehors des itérations ; de ce fait, seules les étapes de diffusion font appel aux opérations de calcul. En conséquence, la complexité spatiale du cryptosystème proposé peut être évaluée à $\Theta(3.M.N)$ avec un processeur Core I7. Cette valeur obtenue est satisfaisante, comparée à d'autres travaux développés dans la littérature ([5] $\Theta(24.M.N)$, [6] $\Theta(4.M.N)$ et [4] $\Theta(100.M.N)$).

3.2.6 Cryptage de plus de deux images

Plusieurs images (≥ 2) peuvent être cryptées par l'algorithme proposé : dans ce cas, les images sources sont combinées en deux grandes images de même taille et chacune d'elle cryptée séparément. En guise d'illustration, les images de la base de données (A-H) sont combinées en deux images (a) et (b) de tailles (1320,1760). Les images chiffrées correspondantes sont présentées à la figure 3.8.

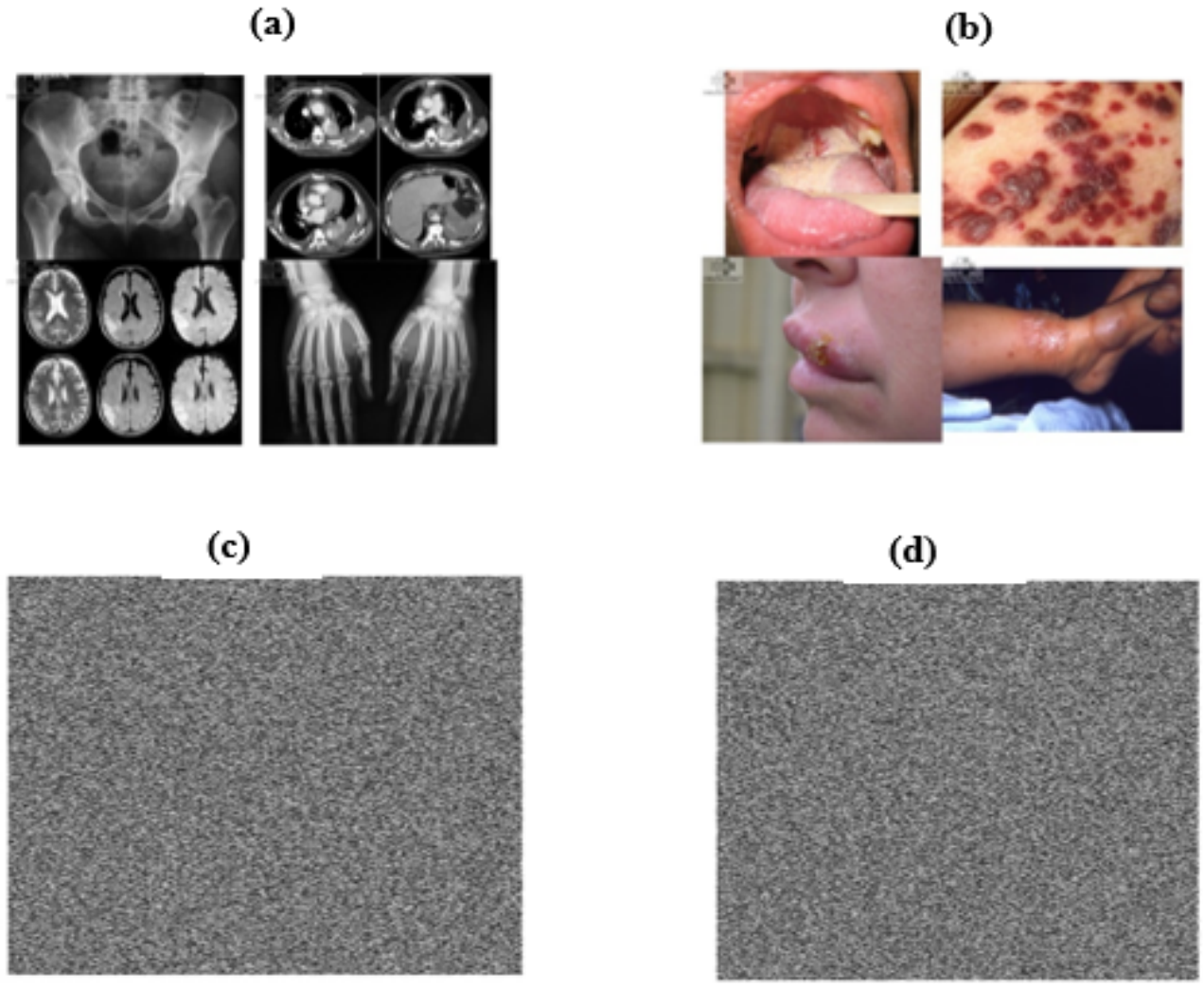


FIGURE 3.8 – Images combinées et chiffrées. (a-b) images combinées, (c-d) images chiffrées hybrides.

Les résultats de tests effectués sur l'image cryptée (c) précédente sont mentionnés dans le tableau 3.11. L'on peut observer que les performances obtenues lors du cryptage de plus de deux images sont meilleures que celles obtenues avec deux images ; également, le temps de

cryptage est satisfaisant pour un nombre de 8 images cryptées. Toutefois, lorsque le nombre d'images à chiffrer augmente, le temps de cryptage va aussi croissant. Il devient donc nécessaire de diminuer la taille d'images combinées par compression avant la phase de cryptage pour réduire le temps de chiffrement.

TABLE 3.11 – Performances du premier algorithme pour le chiffrage de plusieurs images.

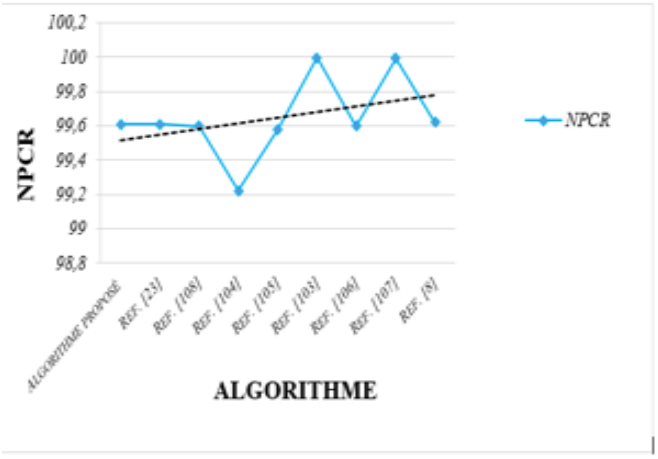
Images	**Tests**	**Algorithme proposé**
Deux images, taille (1320×1760)	Correlation moyenne	0.004
soient 8 images de	Entropie	7.9995
taille (660×880)	NPCR	99.66
	UACI	33.53
	Temps de cryptage (s)	0.9125

3.2.7 Synthèse de l'évaluation de l'algorithme, et discussion

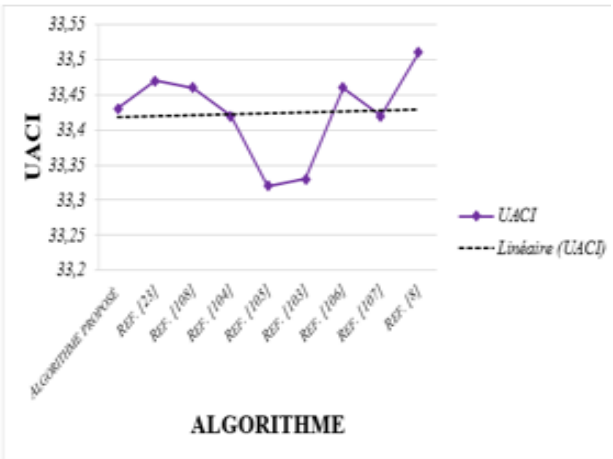
Les performances de l'algorithme proposé comparées à certains récents et meilleurs algorithmes dans la littérature sont illustrées dans le tableau 3.12. Les principales métriques UACI, NPCR, entropie et espace de clé sont illustrés sur la figure 3.9. En comparant l'algorithme proposé à d'autres ([23], [25], [103]–[107]), le nôtre est spécifique, car permet d'assurer la sécurité des données transmises (images médicales) d'un point à l'autre à deux niveaux : au niveau de la clé secrète de chiffrement et au niveau du canal de transmission. Dans ce cryptosystème proposé, le fait de chiffrer les images originales de manière hybride et d'effectuer leur transmission au récepteur par différents canaux si nécessaire, rend l'attaque à force brute sur les images chiffrées impossible. Bien plus, les images cryptées hybrides à transmettre d'un point à l'autre peuvent être envoyées au récepteur par des canaux indépendants et différents, ce qui limite fortement l'action de cryptanalyse sur le système. Etant donné que les images médicales sont sensibles et nécessitent de ce fait un niveau de sécurité élevé lors de leur transmission, le cryptosystème proposé est adapté, car il est simple et garantit un niveau de sécurité élevé. Il est également important de noter que notre approche offre un large espace de clé comparé à ceux des travaux ([23], [103]–[108]). Aussi, il présente des valeurs traduisant une bonne sensibilité la clé ,comparé aux travaux ([23], [104]–[106]), il présente un temps de chiffrement réduit (meilleur que ceux des travaux [106]–[108]), ce qui le rend adapté aux applications à temps réel. De plus, les métriques UACI et NPCR obtenues sont très proches des valeurs standards, et l'entropie a une bonne valeur qui est proche de la valeur idéale qui est 8.

TABLE 3.12 – Comparaison de l'algorithme proposé avec d'autres de la littérature

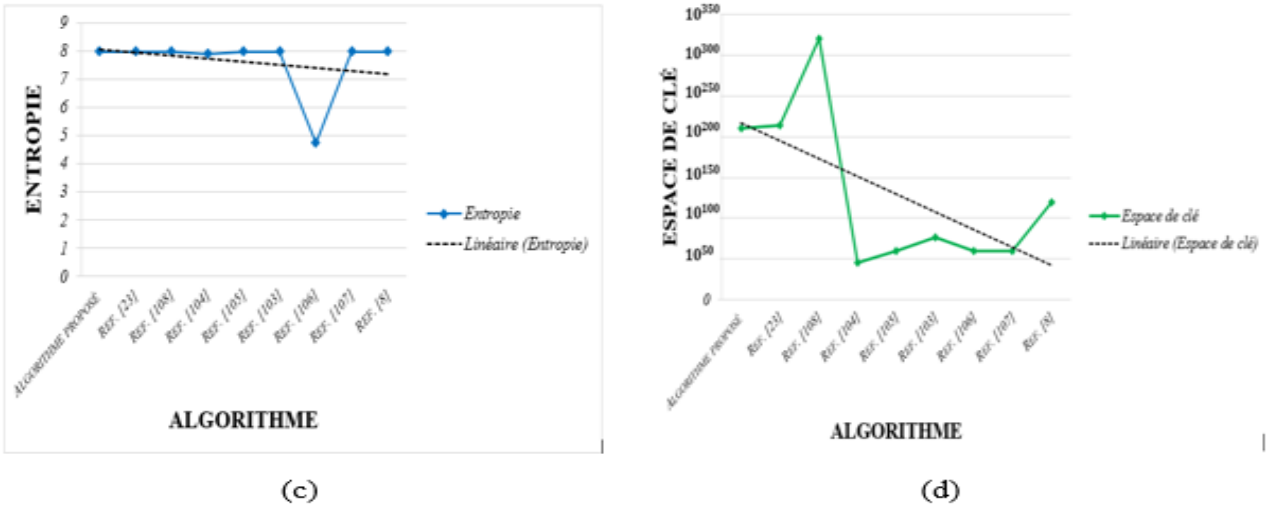
Algorithme	Nbre d'images	Entropie	Cor. Moy.	UACI	NPCR	Temps de cryptage(s)	Espace de clé	Sensibilité de la clé
Algorithme proposé	2 (880 ×660)	7.9994	0.0041	33.43	99.61	0.7316	10^{210}	99.60%
Ref. [23] (2019)	1 (512 ×512)	7.9993	0.0027	33.47	99.61	-	2^{716}	99.30%
Ref. [108] (2019)	1 (512 ×512)	7.9993	0.0044	33.46	99.60	3.3449	3.402 $\times 10^{94}$	99.62%
Ref. [104] (2018)	1 (512 ×512)	7.91	-	33.42	99.22	-	10^{45}	96.20%
Ref. [105] (2018)	1 (512 ×512)	7.9969	0.0025	33.32	99.58	-	10^{60}	95%
Ref. [103] (2018)	1 (512 ×512)	7.9977	-	33.33	99.998	0.4644	2^{256}	-
Ref. [106] (2019)	1 (512 ×512)	4.7453	0.0356	33.46	99.60	1.0152	10^{60}	99.30%
Ref. [107] (2018)	1 (512 ×512)	7.9977	0.0018	33.42	99.992	13.5	10^{60}	-
Ref. [8] (2020)	1 (512 ×512)	7.9994	0.003	33.51	99.62	0.402	10^{120}	-



(a)



(b)



(c) (d)

FIGURE 3.9 – Graphe des métriques principales de l'algorithme comparées à celles de la littérature. (a) UACI, (b) NPCR, (c) entropie, (d) espace de clés.

3.3 Analyse de sécurité de la contribution 2 : « Cryptage d'images par fusion utilisant la transformation discrète en cosinus (DCT) et les générateurs de nombres pseudo-aléatoires »

Dans ce deuxième algorithme proposé, nous utilisons les images médicales (A) à (H) de la base de données (figure 3.2). Dans le processus, quatre images de même taille sont multiplexées en deux grandes images tel que décrit au chapitre 2 (figure 2.21).
Les images A, B, C et D sont combinées en l'image I_2 , et E, F, G et H sont combinées en l'image I_2. L'avantage de combiner les images cibles en deux images est de maximiser le nombre d'images à chiffrer sans altérer la qualité d'images reconstruites à la fin du processus de chiffrement.
Les paramètres de la clé que nous avons utilisés pour effectuer les tests sont les suivants : $x_{01} = 0.351482953177765$; $x_{02} = 0.972970074275508$; $p_1 = 0.488242173292221$; $p_2 = 0.4092407721310 21$; $x_{p1} = 0.363606938668312$; $x_{p2} = 0.890363879273465$; $r_{p1} = 4.841585120587438$; $r_{p2} = 4.7 38149127386060$; $\alpha = 6.187$. La taille du filtre utilisé est déterminée par la relation 2.29 et vaut $(M', M') = (220, 220)$, soit un rapport de compression de 0.75. Les images à chiffrer, toutes de taille 880×660 ont été ajustées afin de les adapter à la taille du filtre.
La taille du filtre utilisé ainsi que le nombre d'images N à multiplexer constitue des paramètres

additionnels de la clé.

La figure 3.11 présente l'image multiplexée I_1 avant et après application de la DCT inverse.

FIGURE 3.10 – Images combinées. (i) images (A)-(D) fusionnées, (j) image (i) après application de la DCT inverse.

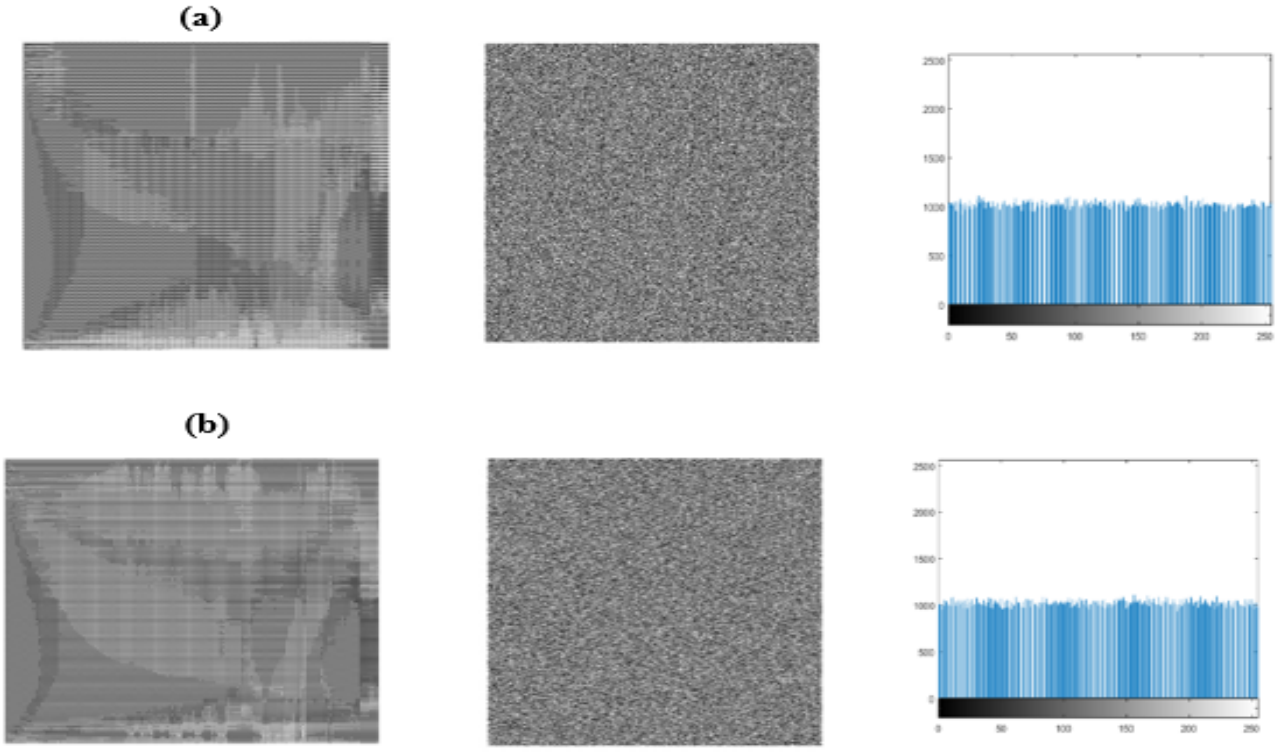


FIGURE 3.11 – Images multiplexées I_1 et I_2 cryptées et leur histogramme. (a) image I_1 , (b) image I_2

Les différentes images déchiffrées correspondant aux 8 images chiffrées (A-H) sont illustrées à la figure 3.12.

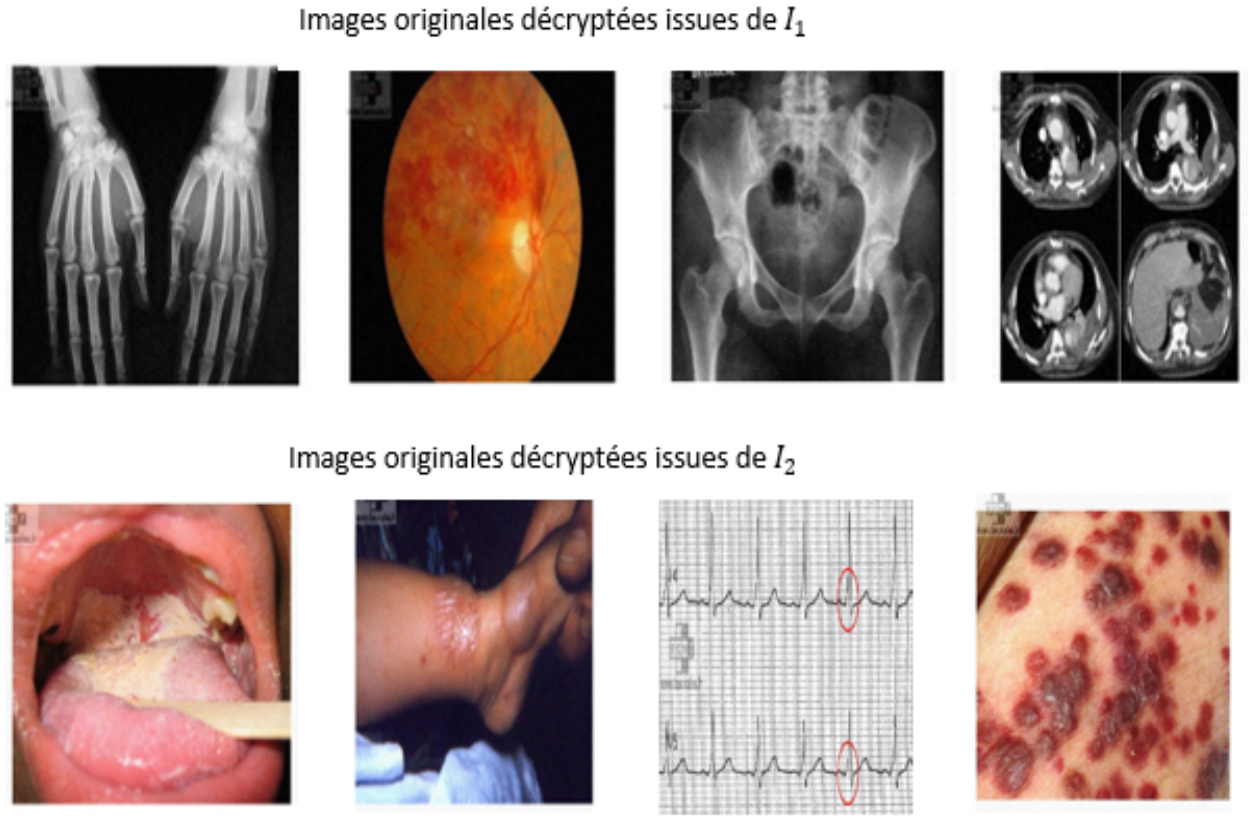


FIGURE 3.12 – Images déchiffrées.

3.3.1 Analyse d'histogramme

La figure 3.11 révèle les histogrammes d'images chiffrées qui présentent une distribution uniforme. Le crypto-système résiste effectivement aux attaques par histogramme.

3.3.1.1 Analyse par corrélation

Les images décryptées présentent la même distribution. Le tableau 3.13 présente les valeurs des coefficients de corrélation de quelques images chiffrées. La moyenne des coefficients de corrélation est de 0.005, qui est proche de zéro. De plus, la figure 3.13 présente la distribution des valeurs de pixels de l'image bassin (880×660) où les pixels sont fortement décorrélés après chiffrement. Ces résultats confirment la robustesse du crypto-système contre les attaques par analyse d'histogramme.

TABLE 3.13 – Coefficients de corrélation de quelques images chiffrées.

Image	Taille	Test	Image en texte clair	Image chiffré
Cameraman	(512 × 512)	HC	0.9314	0.023
		VC	0.9400	0.051
		DC	0.8931	-0.003
Phalange	(880 × 660)	HC	0.9825	-0.008
		VC	0.9990	0.004
		DC	0.9790	0.002
Bassin	(880 × 660)	HC	0.9855	0.002
		VC	0.9983	-0.012
		DC	0.9815	0.006
Thorax	(880 × 660)	HC	0.9210	0.004
		VC	0.9748	0.002
		DC	0.9016	0.009

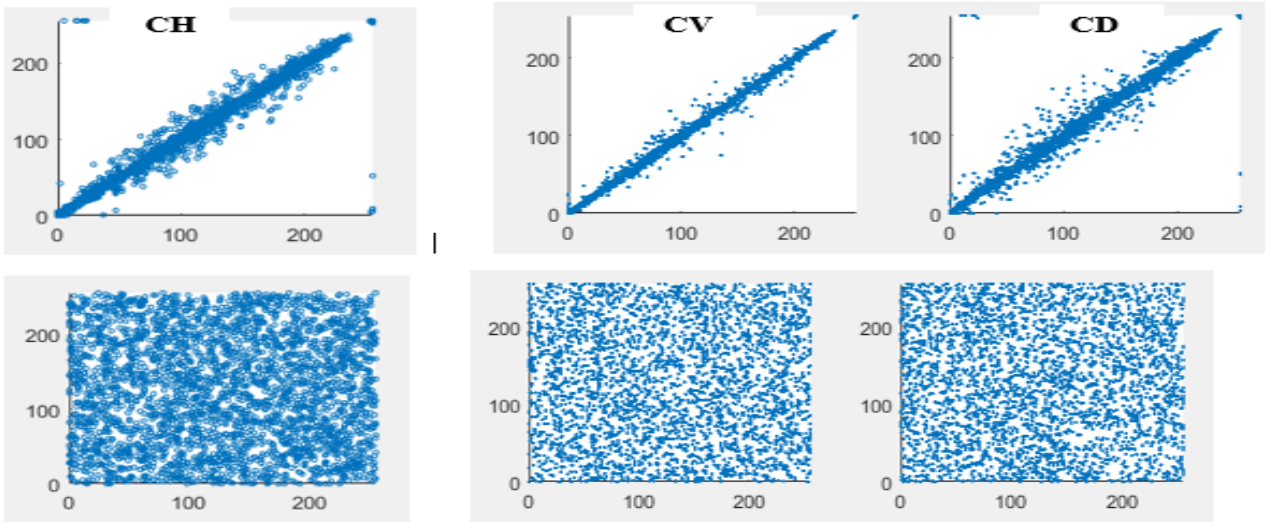


FIGURE 3.13 – Distribution des coefficients de corrélation de l'image bassin (880 × 660) selon les directions horizontale (CH), verticale (CV) et diagonale (CD). (En haut) image originale, (en bas) image chiffrée.

3.3.2 Analyse par entropie de l'information

Le tableau 3.14 présente les valeurs de l'entropie des images chiffrées. La valeur d'entropie est de 7.94 qui est satisfaisante.

TABLE 3.14 – Valeurs de l'entropie des images chiffrées.

Image	**Taille**	**Algorithme proposé**
Image multiplexée I_1	(880×660)	7.9994
Image multiplexée I_2	(880×660)	7.9994

3.3.3 Analyse de la clé

3.3.3.1 Espace de la clé

Le crypto-système proposé comporte 9 paramètres. Les calculs étant effectués avec une précision de 10^{-15} , l'espace de clé est égal à $10^{15\times 9} = 10^{135}$. Cet espace est large pour résister aux attaques à force brute.

3.3.3.2 Sensibilité de la clé.

En modifiant le dernier bit (Least Significant Bit) des paramètres x_{02} , x_{p1} et r_{p1} et en gardant le reste de paramètres inchangés, on obtient respectivement les clés K_2 , K_3 et k_4 (K_1 est la clé originale). Les images I_1 et I_2 chiffrées avec ces clés sont différentes. Le pourcentage de différence entre ces images est présenté dans le tableau 3.15. Nous pouvons relever le fait que la moyenne de différence entre ces images est de 99.65% ; d'où la clé est sensible au moindre changement du bit.

TABLE 3.15 – Pourcentage de différence entre images chiffrées avec différentes clés.

Clé	**Algorithme proposé**
K_1 Vs K_2	99.641
K_1 Vs K_3	99.570
K_1 Vs K_4	99.754

3.3.4 Temps d'exécution

Le temps pour chiffrer 8 images simultanément par le crypto-système proposé est illustré dans le tableau 3.16. La moyenne de temps est de 0.27389 s pour des images de taille 512×512 (images (e)-(l) de la base de données). En comparant cette performance avec celle de certains algorithmes du même types rencontrés dans la littérature, nous constatons que l'algorithme de cryptage proposé est rapide, et bien adapté pour chiffrer plusieurs images numériques simultanément.

TABLE 3.16 – Comparaison du temps de chiffrement avec d'autres algorithmes

Nombres d'images	Algorithme proposé	[56]	[57]	[109]
08 ou 09 Taille (512×512)	0.27389	0.7103	0.191	11.66
08 Taille (880×660) Image médicales (A-H)	0.3582	-	-	-

3.3.5 Qualité des images reconstruites

Dans la plupart des crypto-systèmes par fusion opérant dans le domaine spectral, lorsque le nombre d'images à chiffrer augmente, la qualité d'images déchiffrées se dégrade. Dans le souci de maintenir les images reconstruites aussi proches que les images sources, nous avons regroupé ces dernières en deux grandes images. Afin d'apprécier le pourcentage de dégradation des images cryptées avec ce crypto-système, nous avons calculé l'erreur moyenne quadratique normalisée (NMSE) déterminée par la relation (2.43). Les valeurs obtenues sont notées dans le tableau 3.17, et montrent de faibles valeurs du NSME, ce qui traduit une bonne qualité d'images décryptées (proches des images originales). Le graphe illustrant les valeurs du NMSE en fonction du nombre d'images à chiffer est illustrée à la figure 3.14. Une comparaison est faite avec les résultats obtenus par Aldossari M. [84], (Voir figure 3.15). Il ressort que les valeurs du MSE obtenues avec le cryptosystème proposé sont nettement meilleures que celles de la figure 3.15 pour un même nombre d'images chiffrées.

TABLE 3.17 – Valeur du NMSE en fonction du nombre d'images chiffrées

	Algorithme proposé			[109]	[86]
Nombre d'images à crypter ($N \times 2$) Taille 512×512	4×2	9×2	16×2	9	9
NMSE	8.2×10^{-4}	1.9×10^{-3}	3.76×10^{-3}	4.5799×10^{-3}	6.02×10^{-2}
Nombre d'images à crypter ($N \times 2$) Taille 512×512 images médicales (A-H)	4×2	9×2	16×2	-	-
NMSE	9.7×10^{-4}	2.41×10^{-3}	4.52×10^{-3}	-	-

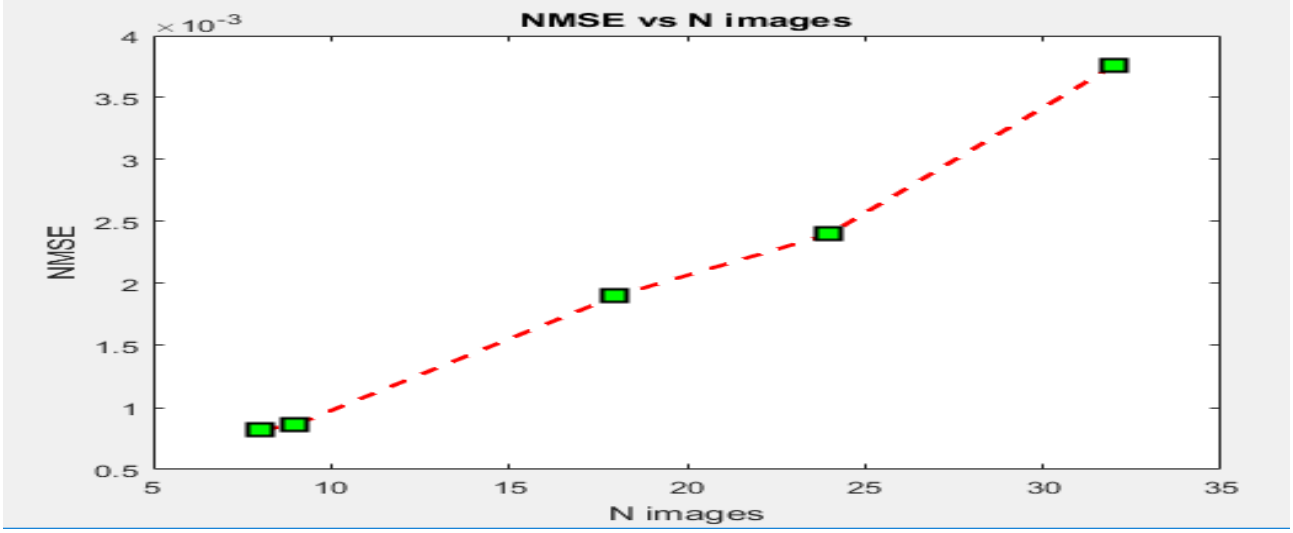


FIGURE 3.14 – Valeurs du NMSE en fonction du nombre N d'images chiffrées par l'algorithme deux.

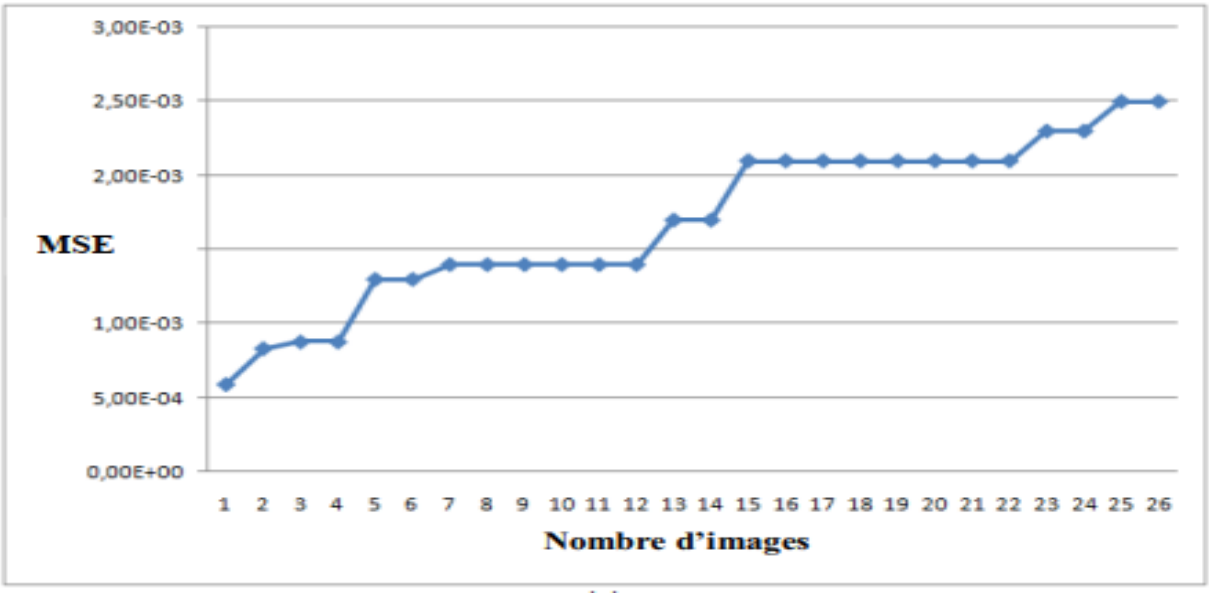


FIGURE 3.15 – Valeurs du MSE en fonction du nombre d'images chiffrées [84].

3.3.6 Analyse différentielle

Les valeurs du NPCR et l'UACI sont relevées dans le tableau 3.18. Les valeurs obtenues sont proches du standard souhaité.

TABLE 3.18 – Valeurs de l'UACI et du NPCR des images chiffrées

Image	Test	
Image multiplexée I_1	NPCR	99.62
(880×660)	UACI	33.51
Image multiplexée I_2	NPCR	99.60
(880×660)	UACI	33.46

3.3.7 Synthèse de l'évaluation du deuxième algorithme, et discussion.

Le deuxième algorithme proposé chiffre plusieurs images de différents types et de même taille pour produire deux images chiffrées contenant toutes les informations des images originales. Le tableau 3.19 fait une comparaison des performances de l'algorithme proposé avec ceux ayant la même structure et chiffrant plusieurs images simultanément. Après observation des résultats du tableau, nous réalisons que l'algorithme proposé chiffre plusieurs images en un temps très réduit, conserve une bonne qualité d'images reconstruites. De plus, les métriques UACI, NPCR, entropie sont proches des meilleures valeurs escomptées. La structure du crypto-système proposée est simple et minimise le nombre d'opérations effectuées. En termes de limitations, lorsque le nombre d'images à chiffrer devient important, l'on observe que la taille des deux images combinées avant le cryptage est très grande. Il devient nécessaire d'effectuer une compression pendant le processus de cryptage afin d'avoir un temps de cryptage satisfaisant.

TABLE 3.19 – Comparaison de l'algorithme proposé avec ceux de la littérature

Tests	**Algorithme proposé**	**[86]**	**[109]**	**[56]**
Espace de clé	10^{135}	10^{210}	10^{90}	10^{60}
Sensibilité de la clé	99.65	-	-	99.72
Moyenne de corrélation	0.005	0.006	0.006	0.004
Entropie	7.9994	7.9989	-	7.9993
NPCR	99.62	99.62	-	99.56
UACI	33.51	33.44	-	33.55
Temps de cryptage (s)	0.27389	0.2386	11.66	0.7103
Nombre d'images chiffrées	8	9	9	9
NMSE	9.7×10^{-4}	6.02×10^{-2}	4.5799×10^{-3}	- -

3.4 Cryptanalyse des crypto-systèmes proposés

Nous avons mené le test de l'attaque à image claire choisie et celle à image chiffrée choisie. Dans le premier cas, l'intrus possède l'image chiffrée C, mais ne dispose pas de la clé de chiffrement. Toutefois, il possède l'image chiffrée C_0 de l'image nulle P_0. Il extrait les sous-clés utilisées pour le chiffrement de l'image par la relation

$$Sk_0^{i,j} = C_0^{i,j} \oplus P_0^{i,j} \tag{3.2}$$

L'opération $(C_0^{i,j} \oplus P_0^{i,j})$ extrait les sous-clés ${Sk_0}^{i,j}$ (William S., 2006). Ainsi, les sous clés obtenues sont utilisés pour recouvrir l'image P par la relation 3.3 suivante

$$P^{i,j} = C^{i,j} \oplus Sk_0^{i,j} \tag{3.3}$$

La figure 3.16-a présente le résultat de l'attaque à image claire choisie de l'image Cameraman (512×512) qui n'a pas connue de succès.
Dans le cas de l'attaque à image chiffrée choisie, l'intrus dispose de l'image C_0 , version cryptée de l'image nulle P_0 . Il cherche à déterminer la séquence de clés $Sk_0^{i,j}$ afin de recouvrir l'image originale P. Ce test a été mené sur l'image Cameraman et n'a pas connu de succès (voir figure 3.16-b).
En somme, après analyse des deux tests principaux de cryptanalyse, il ressort que le premier algorithme proposé résiste aux attaques de cryptanalyse. Après les tests, les mêmes conclusions ont été tirées sur le deuxième crypto-système proposé.

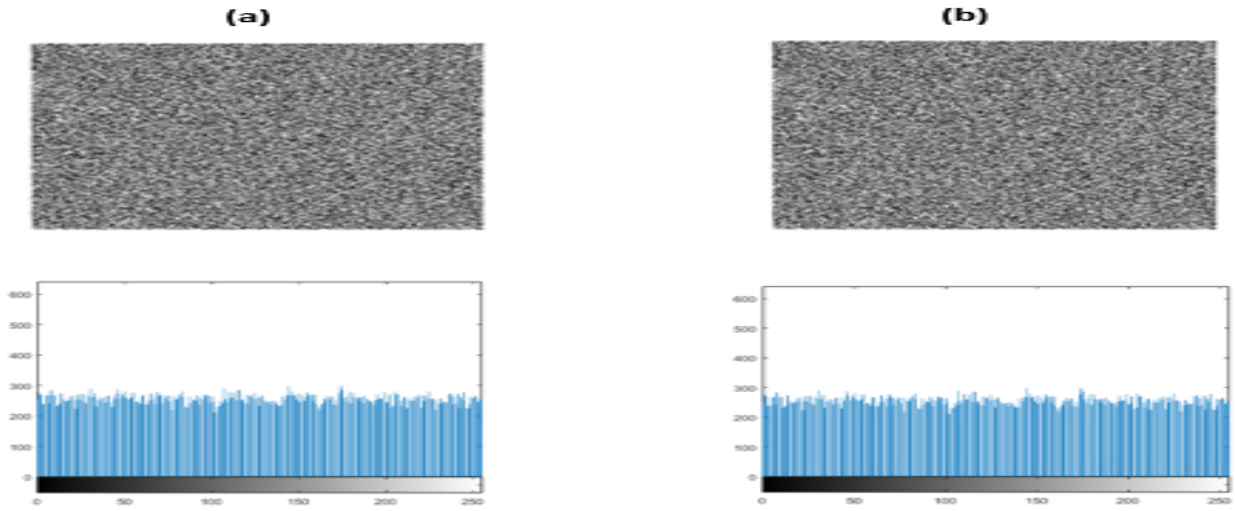


FIGURE 3.16 – Cryptanalyse. (a) attaque à image claire choisie, (b) attaque à image chiffrée choisie de l'image cameraman (512×512).

La figure 3.17 présente respectivement les attaques à image claire choisie et image chiffrée choisie sur l'image œil (660 × 880) chiffrée avec le deuxième algorithme proposé. L'on peut observer que ces attaques sont infructueuses ; par conséquent, le deuxième algorithme proposé résiste bien aux attaques principales de cryptanalyse.

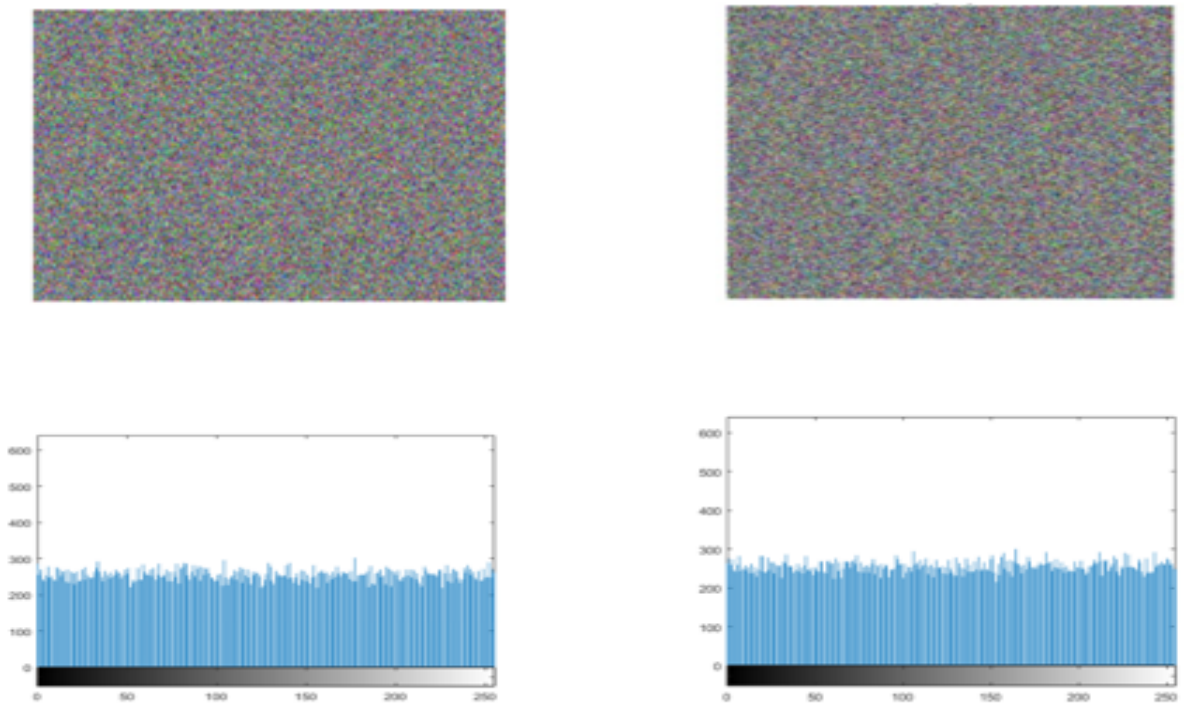


FIGURE 3.17 – Cryptanalyse. (a) attaque à image claire choisie, (b) attaque à image chiffrée choisie de l'image Œil (660 × 880) chiffrée avec le deuxième algorithme.

3.5 Autres tests

3.5.1 Analyse du bruit gaussien

Le bruit gaussien de moyenne nulle et avec diverses valeurs de la variance est appliqué à l'image à chiffrer afin d'apprécier sa résistance au bruit. La fonction utilisée pour ajouter le bruit Gaussien est donnée par la relation 3.4.

$$IMG = imnoise(img^{'}, gaussian^{'}, 0, \text{var}) \tag{3.4}$$

où img et IMG représentent respectivement les données de l'image décryptée et celle décryptée affectée par la bruit ; var represente la variance ayant les valeurs comprises entre 0.01 et 1. Les figures 3.18 et 3.19 montrent les images décryptées affectées par le bruit de variance 0.4, 0.7 et 0.9 respectivement (par le premier et deuxième algorithme proposés respectivement).
Nous observons sur les figures 3.18 et 3.19 qu'il est toujours possible de découvrir le contenu de l'image après effet du bruit gaussien sur ces dernières, même pour la variance égale à 0.9.

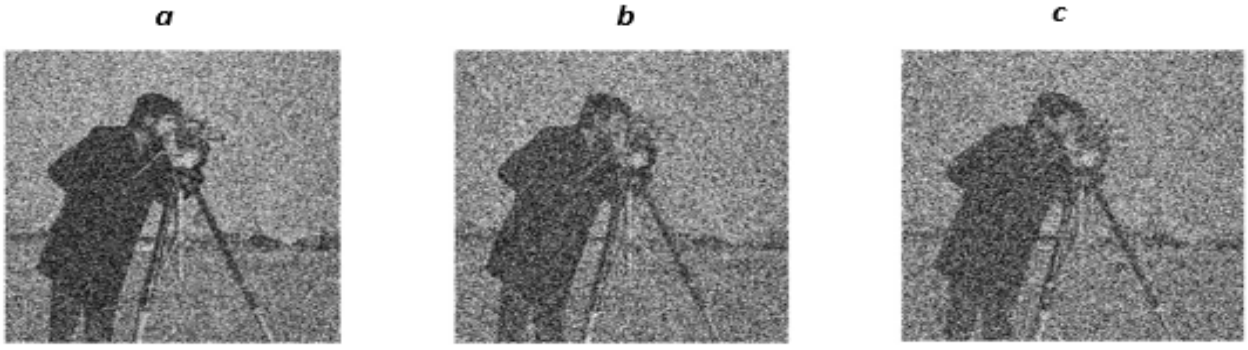


FIGURE 3.18 – Image Cameraman décryptée sous l'effet du bruit Gaussien par l'algorithme 1. (a) var = 0.4. (b) var = 0.7. (c) var = 0.9.

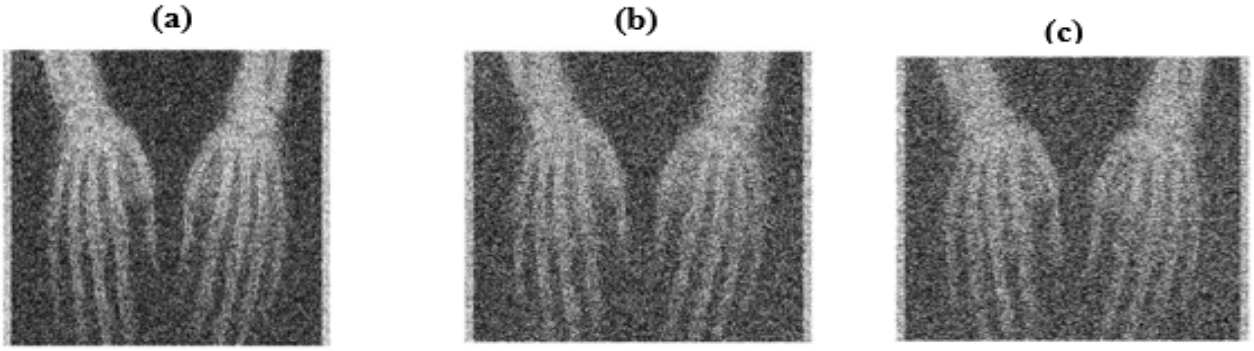


FIGURE 3.19 – Image Phalanges décryptée sous l'effet du bruit Gaussien par l'algorithme 2. (a) var = 0.4. (b) var = 0.7. (c) var = 0.9.

3.5.2 Analyse du bruit d'occlusion

Il arrive souvent que les données perdues lors du processus de transmission et de stockage des informations constituent du bruit additif dans le processus de décryptage. Tel que présenté

à la figure 3.20(a-b), nous considérons que 25% de pixels de l'image cryptée sont perdues (partie en noire sur l'image). Malgré l'effet du bruit additif, l'image décryptée est toujours bien identifiable visuellement (voir figure 3.20-c et d).
En considérant les résultats des deux tests précédents, les deux algorithmes proposés sont résistants à l'effet du bruit.

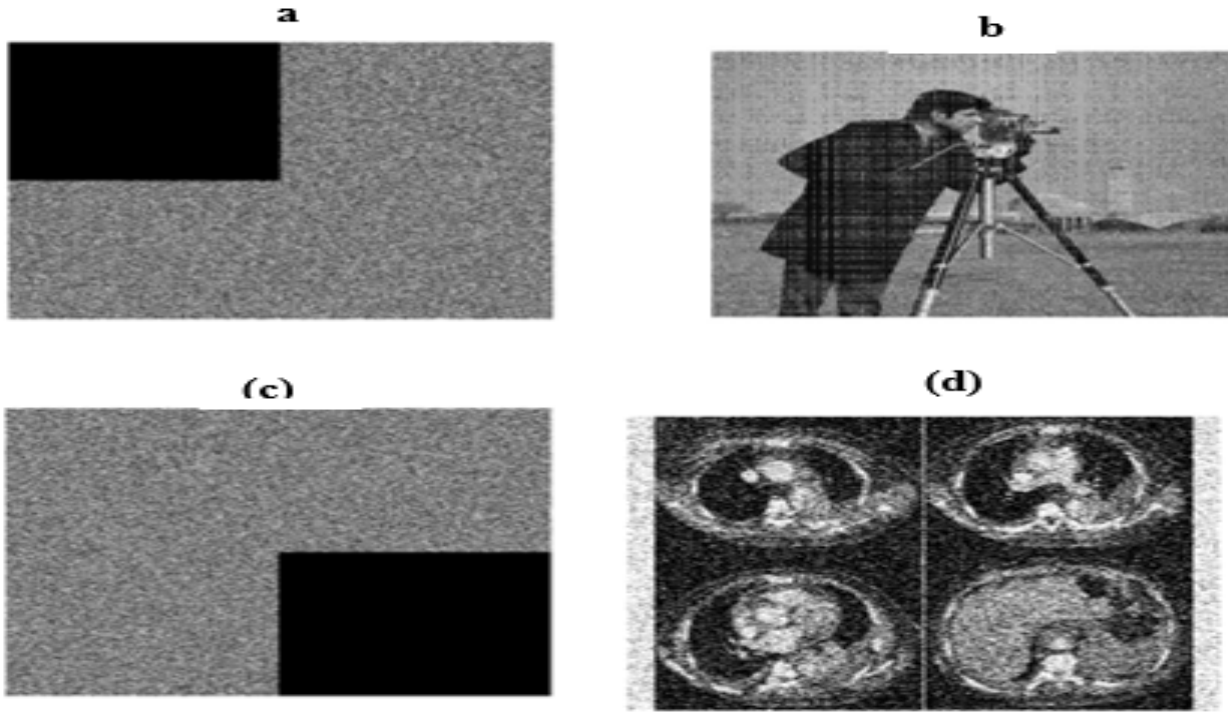


FIGURE 3.20 – Images décryptées sous l'effet du bruit d'occlusion. (a-b) Image Cameraman cryptée et décryptée avec l'algorithme 1, (c-d) Image Thorax cryptée et décryptée avec l'algorithme 2

3.5.3 Analyse comparative des deux crypto-systèmes proposés

Les deux algorithmes proposés ont plusieurs atouts en commun :

- Ils permettent de chiffrer simultanément plusieurs images de différents types ;
- Sont bâtis selon la structure permutation-diffusion ;
- Utilisent des bonnes cartes de données comme générateurs de nombres pseudo-aléatoires ;
- Sont adaptées pour le cryptage de données en temps réel ;
- Sont robustes aux principales attaques de cryptanalyse ;
- Présentent un large espace de clés ;

- Présentent une structure simple et peuvent être implémentées selon une architecture matérielle ;
- Présentent de bonnes performances pour le cryptage d'images médicales.
- L'on n'obtient pas de pertes de données après décryptage avec le premier algorithme.

Malgré les performances obtenues par ces deux algorithmes, chacun d'eux présente tout de même des limitations :

- Dans le processus de cryptage du premier algorithme, pour éviter que la taille de l'image regroupant toutes les images cibles soit très grande, une phase de compression sans perte est nécessaire ; cela permettra d'optimiser le temps de cryptage.
- Pour le deuxième algorithme, le nombre d'images à chiffrer même s'il est grand ne modifie pas la taille des images multiplexées ; toutefois, des efforts peuvent être menés pour minimer davantage l'erreur commise lors du décryptage (MSE).

Le tableau 3.20 présente les performances des deux algorithmes pour un nombre d'images médicales chiffrées de huit (08), de taille (660×880). L'on peut y relever le fait que :

- L'algorithme 1 présente des performances légèrement supérieures à celles du deuxième algorithme (espace de clé, entropie, moyenne de corrélation) ;
- Pour un même nombre d'images à chiffrer, le temps de chiffrement du deuxième algorithme est préférable à celui du premier ;
- La qualité des images décryptées par le premier algorithme est plus proche des images originales chiffrées, comparément au second algorithme où il y'a tout de même de petites pertes de données.

TABLE 3.20 – Performances des deux cryptosystèmes proposés

Tests	**cryptosystème 1 proposé**	**cryptosystème 2 proposé**
Espace de clé	10^{210}	10^{135}
Sensibilité de la clé	99.60	99.65
Moyenne de corrélation	0.0041	0.005
Entropie	7.9994	7.9994
NPCR	99.61	99.62
UACI	33.43	33.51
Temps de cryptage (s)	0.7316	0.27389
Nombre d'images chiffrées	8	8
NMSE	-	9.7×10^{-4}

3.6 Phases d'implémentation du cryptosystème 1 sur la carte STM32F407ZET6

3.6.1 Processus d'implémentation

Dans l'optique d'intégrer dans la carte STM32F407ZET6 le cryptosystème 1, il est question de procédé selon les étapes suivantes :

- Créer un projet à partir de CubeMX, en configurant les ports et autres périphériques de connexion de la carte STM32F407ZET6 ;
- Générer le code source qui sera ouvert sur l'environnement de développement intégré TrueSTUDIO ;
- Par la suite, la compilation et l'exécution du projet ont été réalisées.

Le détail de ces étapes est contenu en annexe du document.
Au regard de l'algorithme décrit à la section 2.9.3, les étapes principales de la phase d'implémentation du cryptosystème sont les suivantes :

i) **Connecter** les composants du système à la carte : la carte mémoire SD, logée dans la carte contient l'image originale à chiffrer qui y a été chargée ; de même, le processus de connexion de l'écran LCD avec la carte est effectué ;

ii) **Charger** les programmes et divers codes : à travers ses ports série USB, les programmes de cryptage, décryptage et autres algorithmes de codage/décodage sont chargés dans la mémoire ROM de la carte STM32F407ZET6.Il est à noter que tous ces programmes sont compilés en langage C et transférés à la carte via le programmateur ST-LINK / V2 A201710 associé à la carte STM32F407ZET6 ;

iii) **Afficher** l'image cryptée : lorsque l'on appuie sur le bouton 1, l'image originale contenue dans la mémoire ROM est lue, puis le programme de cryptage est lancé ; après la durée du chiffrement, l'image cryptée est affichée sur l'écran LCD. Cette image est au même instant nommée et sauvegardée dans la carte mémoire SD ;

iv) **Afficher** l'image décryptée : comme à l'étape 3, en appuyant sur le bouton 2, l'image cryptée est lue dans la carte SD, puis le programme de décryptage est lancé. Après que la durée de décryptage est achevée, l'image décryptée est affichée à l'écran LCD.

3.6.2 Connexion de la carte avec ses périphériques

A cette première phase d'implémentation, nous avons utilisé une image de 128*128 pixels pour des besoins de limitation de la capacité d'affichage de l'écran LCD. Le cryptosystème utilise la même image en entrée des deux algorithmes de chiffrement indépendants avant d'effectuer la phase de mixage.La figure 3.21 présente la mise en marche de la carte connecté à

l'écran LCD. Par ailleurs, l'image originale est affichée sur l'écran LCD en appuyant sur le bouton 1 tel qu'illustré à la figure 3.22. De même, en appuyant sur le bouton 2, l'image déchiffrée peut être affichée sur l'écran LCD.

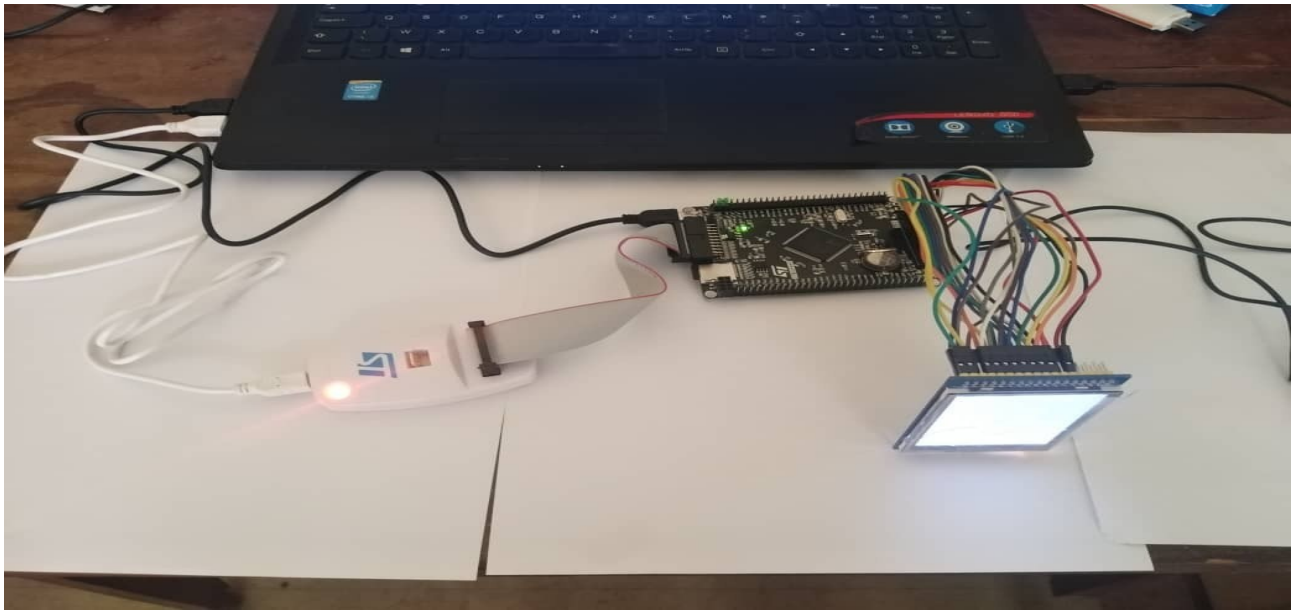


FIGURE 3.21 – Connexion de la carte STM32F407ZET6 avec les autres périphériques

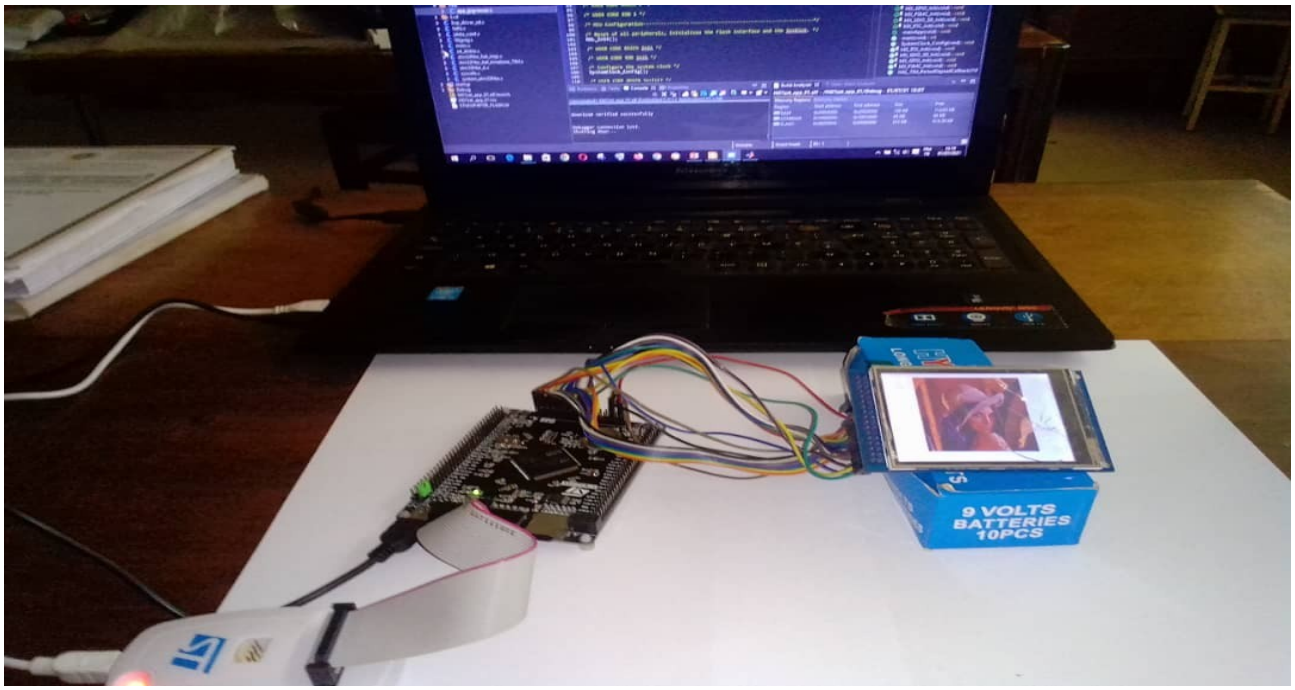


FIGURE 3.22 – Affichage de l'image lena (320x240) sur l'écran LCD

Une fois que l'image cryptée est obtenue, elle doit être transmise au niveau du dispositif récepteur qui peut être, soit un ordinateur, soit un téléphone mobile. Le destinataire appuie sur le bouton 2 pour décrypter l'image. Cette deuxième phase d'implémentation est encore en cours de développement. De manière générale, l'opération de transmission du message crypté est possible en ajoutant à la carte STM32F407ZET6 son module Wifi associé de référence Eps8266 : ce dernier permet ainsi la transmission de l'image chiffrée d'un point d'émission à celui de réception en un temps très bref. Dans le cas où les systèmes d'émission et de réception sont très distants, le mode de transmission doit être modifié ; par exemple on peut connecter les deux systèmes à travers un module de fibre optique.

Conclusion

Tout au long de ce chapitre, nous avons présenté les différents tests d'évaluation des performances des deux crypto-systèmes proposés ainsi que l'analyse de l'implémentation d'un des cryptosystèmes sur la carte intégrée STM32F407ZET6. Il en ressort que les algorithmes évoqués présentent de bonnes performances en termes de robustesse, temps d'exécution, de qualité d'images reconstruites ainsi que de nombre d'images à chiffrer. Également, les valeurs des métriques NPCR, UACI, entropie, comparées à certains algorithmes de la littérature sont proches des valeurs idéales. Le début d'étude de la phase d'implémentation du cryptosystème 1 proposé sur la carte STM32F407ZET6 renseigne de la faisabilité du système. En plus, les tests menés sur les images standards de la littérature ont aussi été congruents sur les images médicales.

Conclusion générale et perspectives

Les travaux de thèse présentés dans ce document traitent du cryptage d'images médicales de différentes tailles par fusion ou mixage d'images. Deux nouveaux algorithmes ont été développés, tous bâtis selon l'architecture permutation-diffusion et utilisent des générateurs de nombres pseudo-aléatoires. En outre, une implémentation du premier algorithme sur la carte STM32F407ZET6 a été faite dans l'optique d'apprécier le fonctionnement en mode réel. Les deux crypto-systèmes proposés répondent au besoin de transmission sécurisée et simultanée de grandes quantités d'images, sans cesse croissantes dans les applications multimédia et autres domaines, notamment en télémédecine. Nous avons réalisé qu'il était possible de chiffrer les images par fusion ou mixage d'images. Cette dernière approche présente les atouts de combiner plusieurs images en une, en minimisant le nombre de données à transmettre et en limitant les pertes d'informations. Bien plus, au travers de cette approche, la transmission sécurisée des données d'un point à l'autre est rehaussée. Certaines techniques de cryptage basées sur la fusion ont été étudiées ; il ressort que la plupart d'elles effectue la fusion dans le domaine spectral et entraine une dégradation de la qualité d'images reconstruites. Pour celles qui procèdent par crypto-compression, les résultats des tests ne permettent pas d'obtenir un bon compromis entre robustesse, rapport de compression et qualité d'images décryptées. Ces failles trouvent des éléments de solution avec les deux algorithmes que nous proposons dans nos travaux. Le premier algorithme possède une structure simple et effectue la fusion de deux ou plusieurs images dans le domaine spatial. L'approche hybride utilisée dans le crypto-système contribue à rendre les images chiffrées robustes et dépendantes les unes des autres. Par ailleurs, sans la connaissance de la clé de chiffrement, il est difficile de casser l'algorithme implémenté. Au fait, ce cryptosystème assure la protection des données à deux niveaux : au niveau de la clé secrète et dans le canal de transmission. Dans l'optique de chiffrer plusieurs images en réduisant la taille de l'image transmise, comme c'est le cas souhaité lorsque la fusion est effectuée dans le domaine spatial, le deuxième algorithme a été proposé. Nous avons à cet effet réussi à fusionner un ensemble d'images à deux niveaux, respectivement dans les domaines spectral et spatial. Les propriétés de la DCT nous ont permises de fusionner un grand nombre d'images cibles sans qu'elles ne soient dégradées lors de la phase de décryptage.

Les tests de sécurité des deux algorithmes ont été menés sur les images standards de la littérature, puis principalement sur des images médicales. Les performances obtenues suite aux

différentes analyses et tests révèlent la robustesse, la rapidité, la bonne qualité d'images déchiffrées ainsi que la possibilité de chiffrer un grand nombre d'images. En faisant une comparaison avec certains meilleurs algorithmes développés récemment, nos résultats sont satisfaisants. Les valeurs des métriques, entropie, NMSE, UACI, NPCR, coefficient de corrélation, temps de chiffrement justifient à suffisance les bonnes performances des algorithmes proposés. L'étude du début de la phase d'implémentation du premier algorithme sur la carte intégrée STM32F407ZET6 s'est montrée rassurante et permet d'envisager la possibilité d'intégration des algorithmes utilisant l'approche de mixage pour des besoins de transmission sécurisée des données. En particulier, un tel système de transmission sécurisée des données à carte intégrée est adapté dans le domaine de la télémédecine

En guise d'impact, les algorithmes proposés sont une contribution importante parmi les algorithmes de chiffrement basés sur la fusion déjà existants. Ces algorithmes présentent un large espace de clés qui répond au problème d'espace de clés réduit de certains algorithmes cryptanalysés. Le premier cryptosystème en particulier assure de par sa structure un double niveau de sécurité à travers la clé secrète, mais aussi dans le canal de transmission. De plus, ces algorithmes permettent de crypter en un temps réduit une grande quantité d'images en assurant un bon compromis entre robustesse, rapidité d'exécution, quantité d'images à transmettre et qualité d'images décryptées. Au regard des atouts cités précédemment, les techniques de cryptage par fusion ou mixage d'images sont prometteuses en matière de sécurité de données, sont implémentables dans des architectures matérielles, et devraient être explorées plus en profondeur.

Perspectives

Les algorithmes proposés dans cette thèse peuvent être améliorés ou étendus vers d'autres voies d'implémentation. Conséquemment, nous suggérons et envisageons les possibilités suivantes :

- Développer des algorithmes de fusion avec une structure simple adaptée aux vidéos assez présentes dans les canaux de communication.
- Continuer et finaliser l'implémentation du cryptosystème 1 sur la carte STM32F407ZET6 pour utilisation dans les systèmes embarqués. Également mener cette opération sur une carte FPGA et effectuer une analyse comparative des performances du système suivant ces deux approches.
- Développer une nouvelle approche de fusion basée sur d'autres critères d'images sources, par exemple des critères de segmentation ou de couleur.
- Proposer des techniques de cryptage de fusion utilisant les relations mathématiques itératives, afin d'améliorer celles existantes par rapport au temps d'exécution.
- Proposer des algorithmes de chiffrement basés sur la fusion combinant l'holographie, le tatouage et la cryptographie ;
- Améliorer le premier algorithme proposé selon la structure crypto-compression, car ce

dernier permet de minimiser la taille de données à transmettre et de limiter le nombre d'opérations effectuées par le processeur.

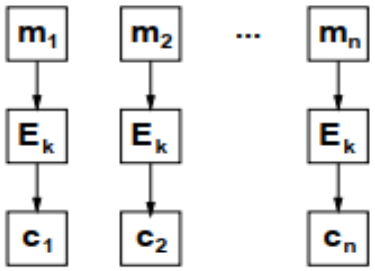
- Utiliser les cartes chaotiques à signaux continus pour effectuer l'implémentation des cryptosystèmes sur les cartes intégrées afin d'améliorer la précision des performances du système.

Annexe

Différents modes de chiffrement en bloc (section 1.4.1.2)

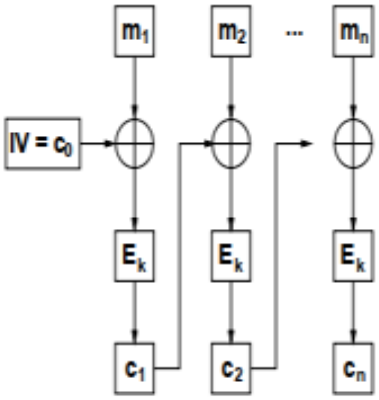
1. **Le mode ECB : Electronic Code Book**
2. **Le mode CBC : Cipher Bloc Chaining**

$$c_i = E_k(m_i) \qquad (1.1)$$



(a) Chiffrement en mode ECB

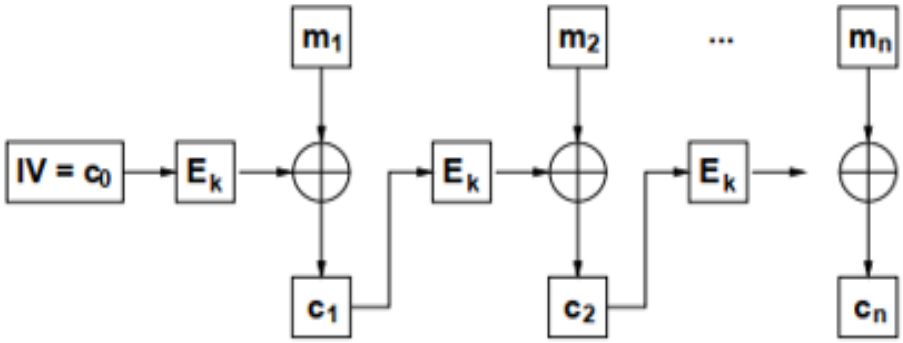
$$c_i = E_k(m_i \oplus c_{i-1}) \qquad (1.2)$$



(b) Chiffrement en mode CBC

3. Le mode CFB : Cipher FeedBack

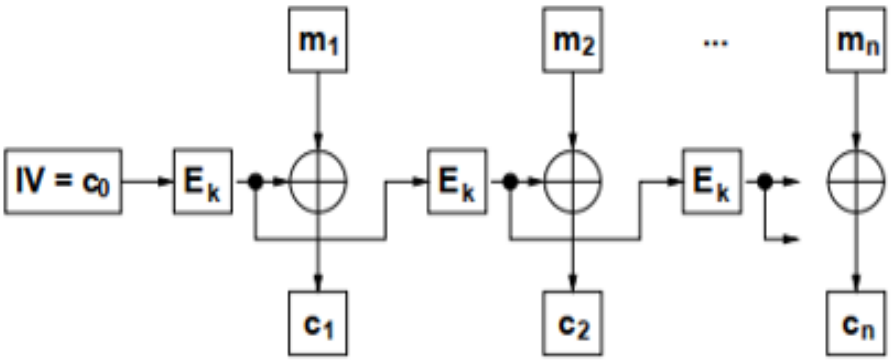
$$c_i = m_i \oplus E_k(c_{i-1}) \tag{1.3}$$



(c) Mode de chiffrement CFB

4. Le mode OFB : Output FeedBack

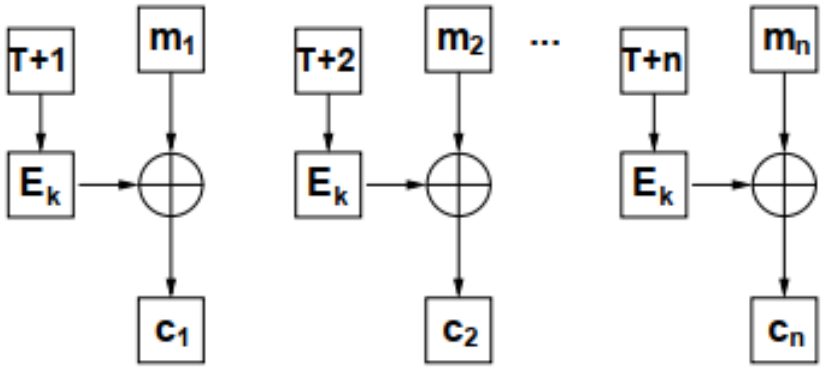
$$z_0 = c_0 \; ; \; z_i = E_k(z_{i-1}) \; ; \; c_i = m_i \oplus z_i \tag{1.4}$$



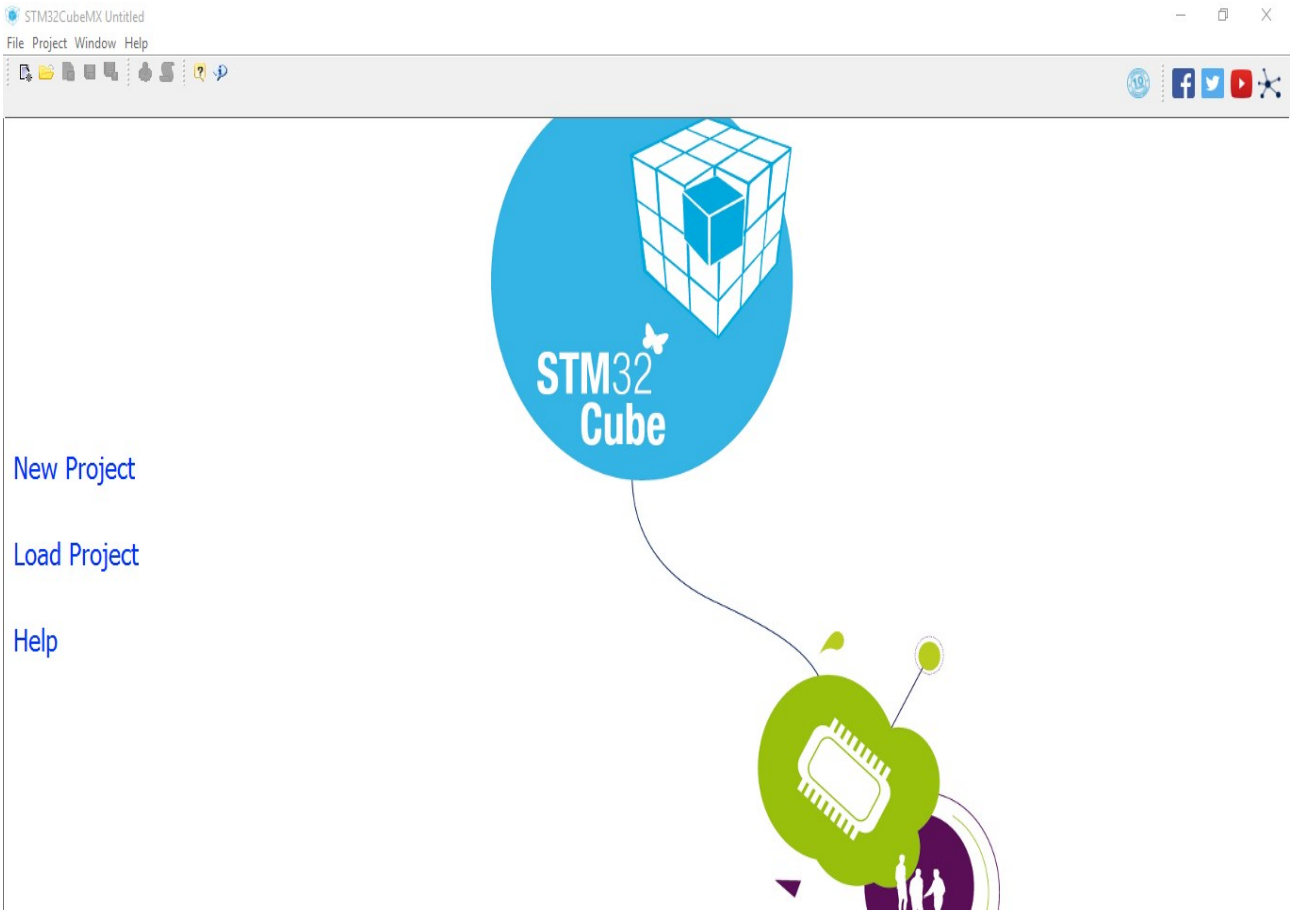
(d) Mode de chiffrement OFB

5. Le mode CTR : Counter-mode encryption

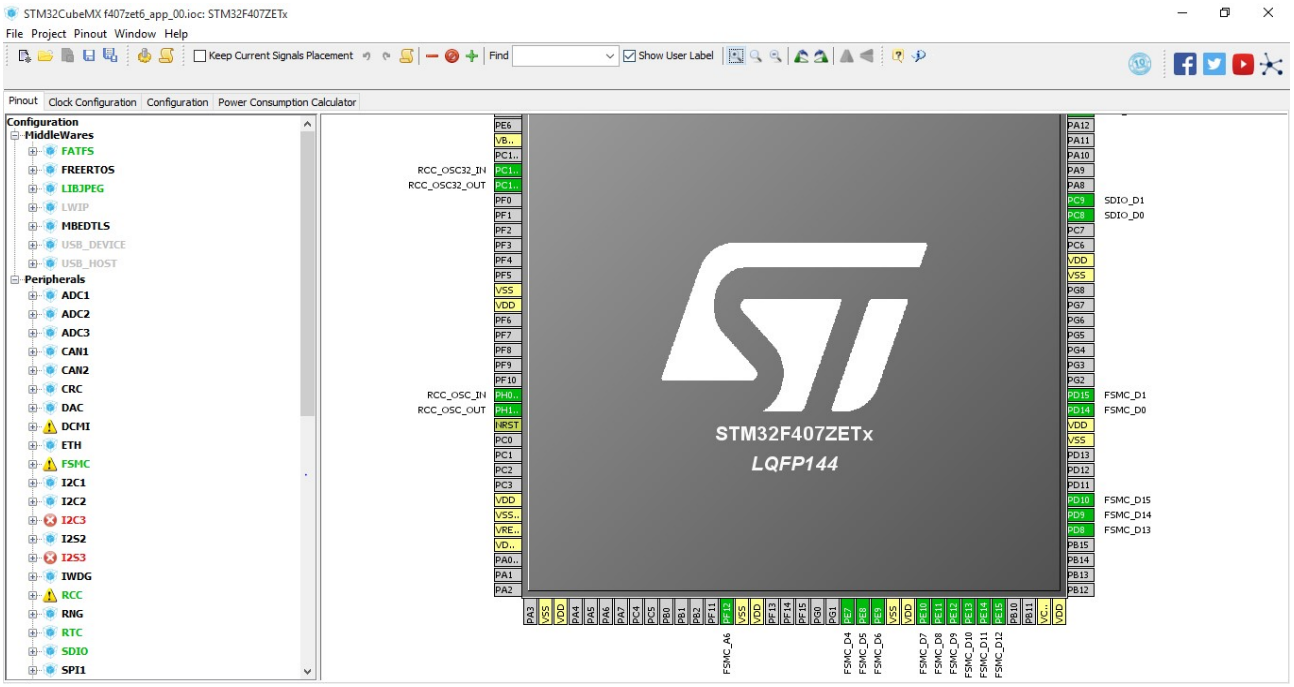
$$c_i = m_i \oplus E_k(T+i) \tag{1.5}$$



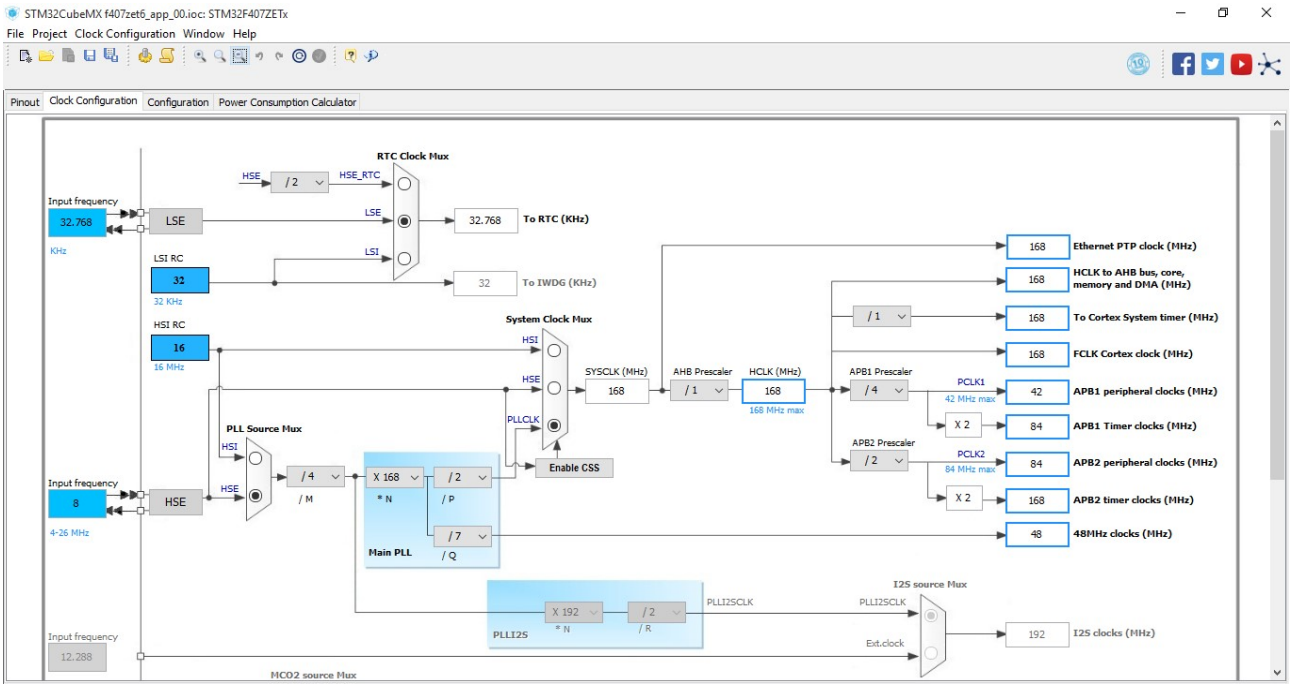
(e) Mode de chiffrement CTR



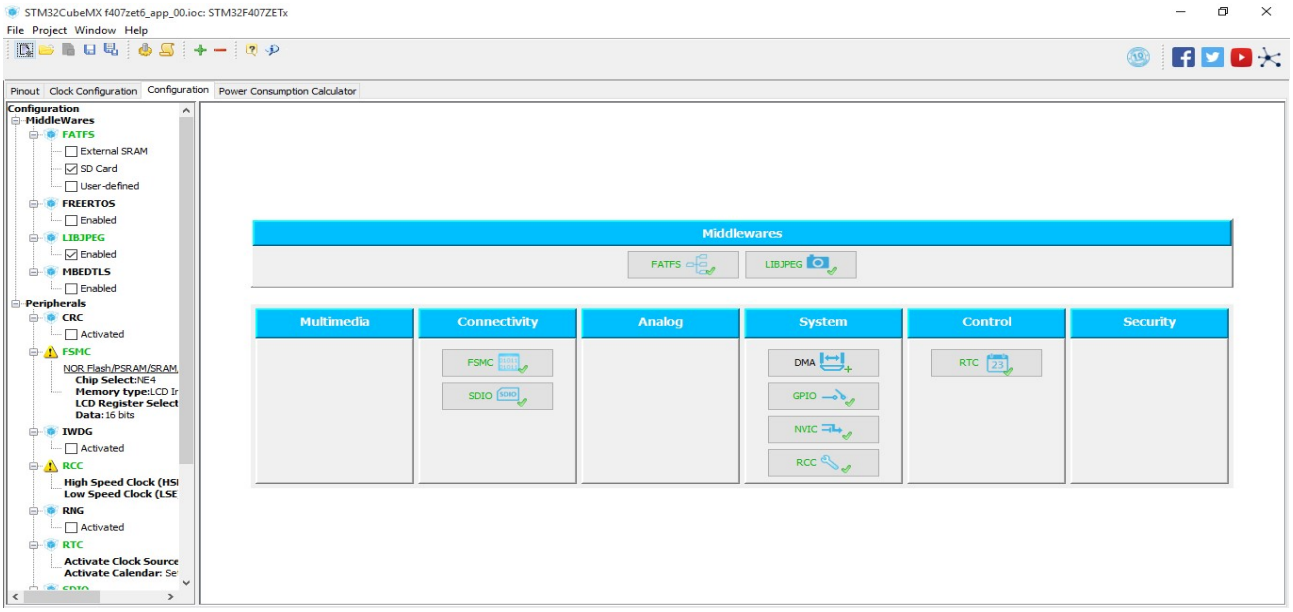
La Fenêtre Principale de STM32CubeMX



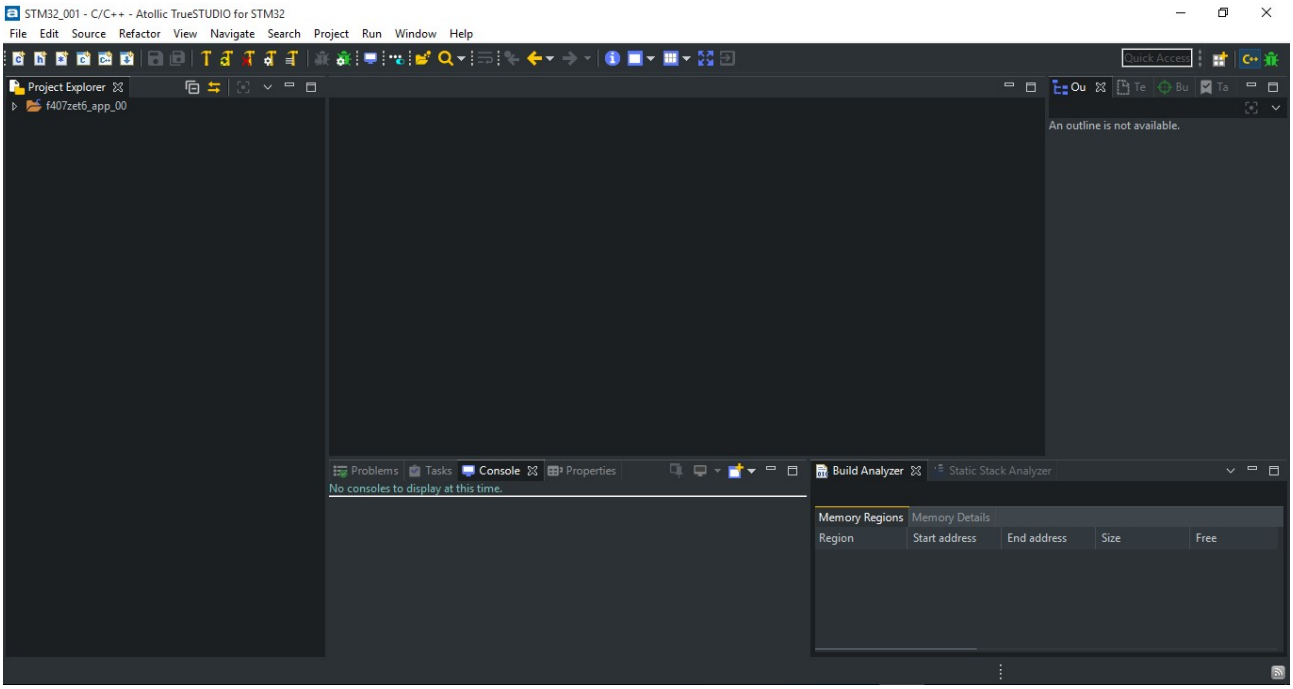
Paramétrage des broches de sorties



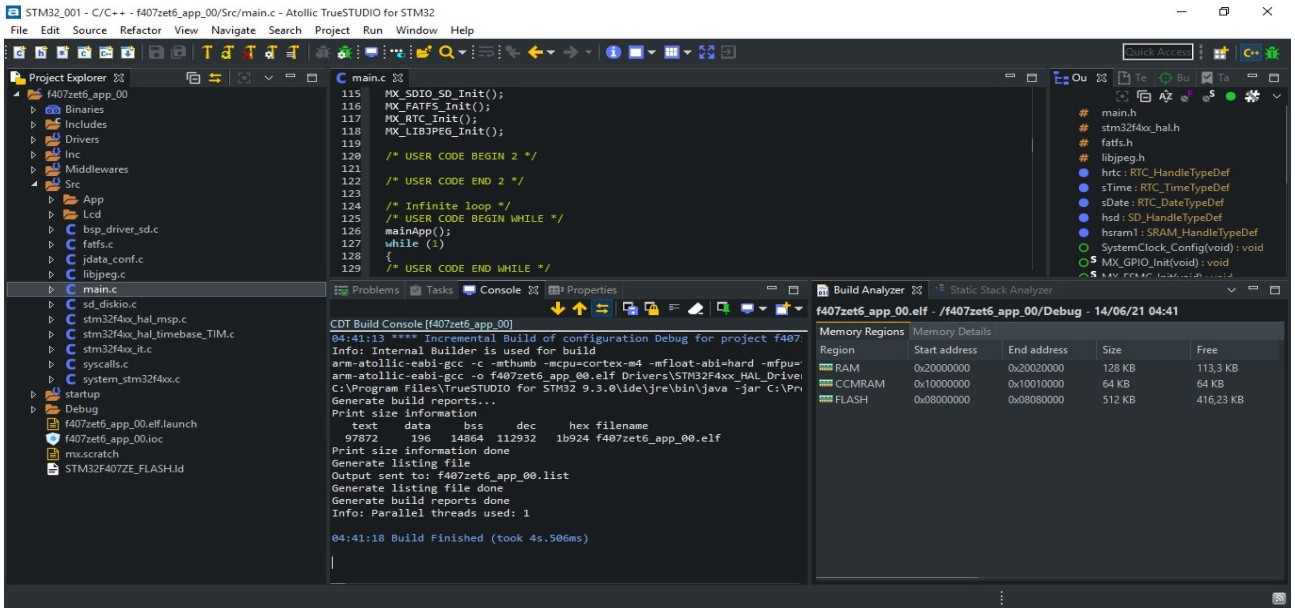
Configuration de l'horloge



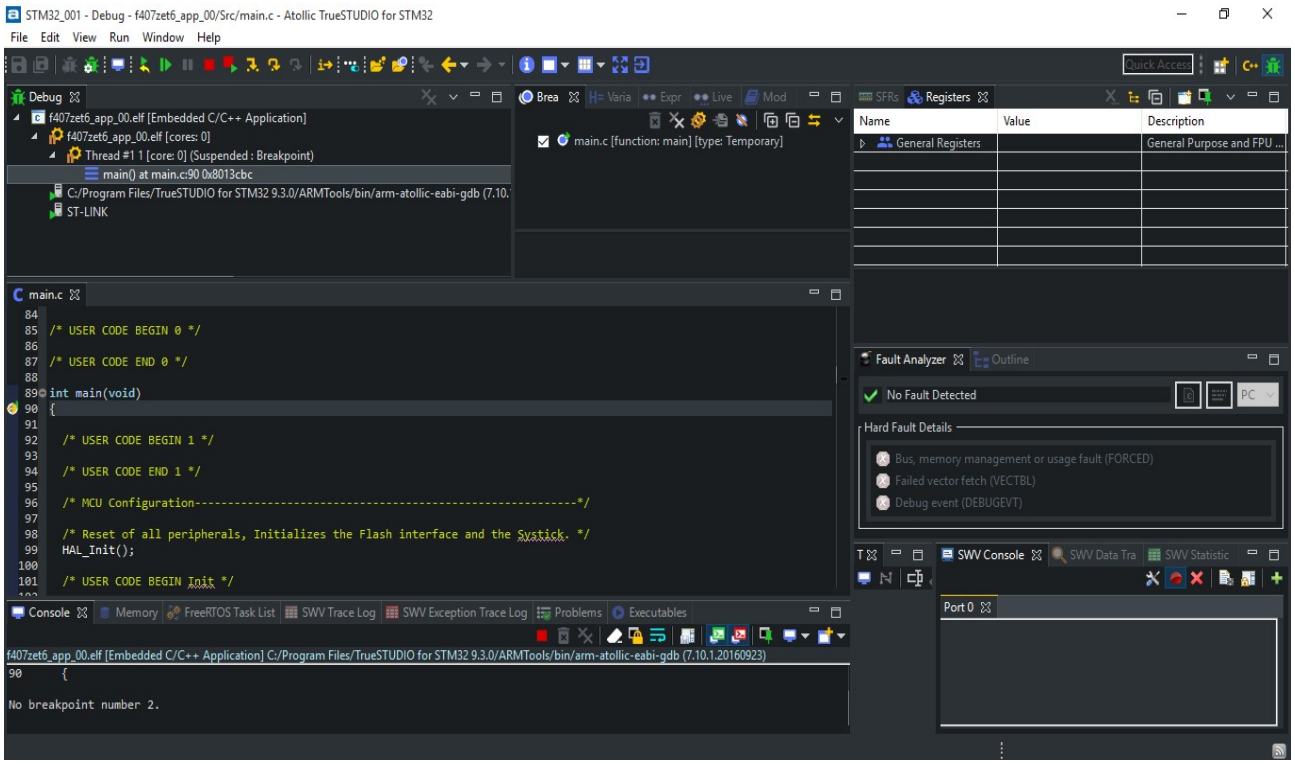
Fenêtre d'activation de la FATFS et du LIBJPEG



Interface de Programmation de TrueSTUDIO



Présentation de L'interface après compilation du projet



Présentation de l'interface TrueSTUDIO lors du débogage du projet

Étapes de mise en oeuvre du projet.

1. Créer un projet pour CubeMX

- réglage du RCC (Résonateur cristal/céramique) ;
- réglage du débogage (SYS / fil série ou Trace assyn Sw) ;
- réglage de la source de la base de temps ;
- réglage du FSMC (sélection de la puce, type de mémoire = LCD, sélection du registre Lcd, données = 16 bits) ;
- réglage du mode SDIO, activation de la FATFS, FATFS : USE_LFN, MAX_SS = 4096, FS_LOCK = 5, RTC activé ;
- si JPG : activation du LIBJPEG ;
- configuration de l'horloge ;
- paramètres du projet : nom du projet, toolchain = truestudio, stack size = 0x800 ;
- générer le code source.

2. Truestudio

- ouvrir les projets depuis le système de fichiers (seulement Truestudio) ;
- ouvrir main.c ;
- ajouter USER CODE BEGIN PFP : void mainApp(void) ; ;
- ajouter le code utilisateur BEGIN WHILE : mainApp() ; ;
- ajouter USER CODE BEGIN Includes (#include "stm32f4xx_hal.h") ;
- ajouter 2 nouveaux dossiers pour le dossier Src (App, Lcd) ;
- copier le(s) fichier(s) de App vers App ;
- copier 4 ou 7 fichiers de Drivers vers Lcd (lcd.h, bmp.h, stm32_adafruit_lcd.h c, si touch : ts.h, stm32_adafruit_ts.h c) ;
- copier le dossier Fonts dans le dossier Lcd ;
- copier le pilote .io dans le dossier Lcd (lcd_io_FSMC16.h c) ;
- copier le pilote lcd dans le dossier Lcd (ili9341.h /c) ;
- configuration du fichier d'en-tête du pilote io (paramètres des broches, paramètres de vitesse, etc...) ;
- configuration de l'écran LCD (orientation, écran tactile) ;
- ajouter le chemin d'inclusion : Src/Lcd ;
- paramétrer les options de compilation (Activer la compilation parallèle, optimisation) ;
- compiler, exécuter ...

Algorithme de configuration du pilote de l'afficheur LCD ILI9341 (16 bits)

```
#define LCD_RST                 X, 0

#define LCD_BL                  B, 15
#define LCD_BLON                0

#define LCD_ADDR_BASE           0x6C000000 // FSMC_NE4
#define LCD_REGSELECT_BIT       6          // FSMC_A6

/***** If not used DMA : *****/
#define LCD_DMA                 0, 0, 0, 0

/***** If used DMA : *****/
#define LCD_DMA                 2, 7, 7, 1    // MEM to MEM
```

Liste des publications

1. **L. M. H. Yepdia, A. Tiedeu, and G. Kom**, A Robust and Fast Image Encryption Scheme Based on a Mixing Technique. *Security and Communication Networks,* , Article ID 6615708, 17 pages (2021), https ://doi.org/10.1155/2021/6615708
2. **L. M. H. Yepdia, A. Tiedeu**, Secure transmission of medical image for telemedicine. *Sens Imaging* 22, 17 (2021). https ://doi.org/10.1007/s11220-021-00340-8

Chapitre de livre

1. **L. M. H. Yepdia, A. Tiedeu, and Z. Lachiri**, Multiple-Image Fusion Encryption (MIFE) Using Discrete Cosine Transformation (DCT) and Pseudo Random Number Generators. Multimedia Information Retrieval, DOI : http ://dx.doi.org/10.5772/intechopen.92369, 2020, pp. 1-18.

Bibliographie

[1] E. Edition, E. Edition, O. Systems, S. Edition, B. D. Communications, and S. Edition, *The Willian Stalling Books on Computer Data and Computer Communications,* EIGHTH EDITION.

[2] **M. Zhou and C. Wang** , "A novel image encryption scheme based on conservative hyperchaotic system and closed-loop diffusion between blocks". *Signal Processing,* p. 107484, 2020, doi : 10.1016/j.sigpro.2020.107484.

[3] **A. Awad**, "A New Chaos-Based Cryptosystem for Secure Transmitted Images," *IEEE Transactions on Computers, Institute of Electrical and Electronics Engineers,* vol. 99, pp.1, 2011.

[4] **I. S. Sam, P. Devaraj, and R. S. Bhuvaneswaran**, "An Efficient Quasigroup Based Image Encryption Using Modified Nonlinear Chaotic Maps," *Sens. Imaging,* vol. 15, no. 1, 2014, doi : 10.1007/s11220-014-0092-x.

[5] **C. Y. Song, Y. L. Qiao, and X. Z. Zhang**, "An image encryption scheme based on new spatiotemporal chaos," *Optik (Stuttg).,* vol. 124, no. 18, pp. 3329–3334, 2013, doi : 10.1016/j.ijleo.2012.11.002.

[6] **S. M. Seyedzadeh, B. Norouzi, M. R. Mosavi, and S. Mirzakuchaki**, "A novel color image encryption algorithm based on spatial permutation and quantum chaotic map," *Nonlinear Dyn.,* vol. 81, no. 1–2, pp. 511–529, 2015, doi : 10.1007/s11071-015-2008-2.

[7] **S. M. Ismail, L. A. Said, A. G. Radwan, A. H. Madian, and M. F. Abu-elyazeed**, "Generalized double-humped logistic map-based medical image encryption," *J. Adv. Res.,* vol. 10, pp. 85–98, 2018, doi : 10.1016/j.jare.2018.01.009.

[8] **Y. P. K. Nkandeu, J. R. Mboupda Pone, and A. Tiedeu**, *Image Encryption Algorithm Based on Synchronized Parallel Diffusion and New Combinations of 1D Discrete Maps,* vol. 21, no. 1. Springer US, 2020.

[9] **X. Zhang and Z. Zhao**, "Chaos-based image encryption with total shuffling and bidirectional diffusion," *Nonlinear Dyn.,* vol. 75, no. 1–2, pp. 319–330, 2014, doi : 10.1007/s11071-013-1068-4.

[10] **I. F. Elashry et al.**, "Efficient chaotic-based image cryptosystem with different modes of operation," *Multimed. Tools Appl.,* vol. 79, no. 29–30, pp. 20665–20687, 2020, doi : 10.1007/s11042-019-08322-5.

[11] **S. Koppu and V. M. Viswanatham**, "A Fast Enhanced Secure Image Chaotic Cryptosystem Based on Hybrid Chaotic Magic Transform," Modelling and Simulation in Engineering, vol. 2017., ID07470204.

[12] **Y. Abanda and A. Tiedeu**, "Image encryption by chaos mixing," *IET Image Process.,* vol. 10, no. 10, pp. 742–750, Oct. 2016, doi : 10.1049/iet-ipr.2015.0244.

[13] **J. S. A. Eyebe, J. Y. Effa, S. L. Sabat, and M. Ali** "Commun Nonlinear Sci Numer Simulat A fast chaotic block cipher for image encryption," *Commun. Nonlinear Sci. Numer. Simul.,*vol. 19, no. 3, pp. 578–588, 2014, doi : 10.1016/j.cnsns.2013.07.016.

[14] **M. A. Ben Farah, R. Guesmi, A. Kachouri, and M. Samet**, "A novel chaos based optical image encryption using fractional Fourier transform and DNA sequence operation," *Opt. Laser Technol.,* vol. 121, no. April 2019, p. 105777, 2020, doi : 10.1016/j.optlastec.2019.105777.

[15] **Y. A. S. Hi, Y. O. L. Iu, W. E. I. S. Heng, D. A. B. O. Z. Hu, and J. I. W. Ang**, "rotations of a random phase mask with spatially incoherent illumination," Optics Express, vol. 27, no. 18, pp. 26050–26059, 2019.

[16] **X. Yuan, L. Zhang, J. Chen, K. Wang, and D. Zhang**, "Multiple - image encryption scheme based on ghost imaging of Hadamard matrix and spatial multiplexing," *Appl. Phys. B,* vol. 125, no. 9, pp. 1–13, 2019, doi : 10.1007/s00340-019-7286-9.

[17] **N. Zhou, J. Yang, C. Tan, S. Pan, and Z. Zhou**, "Double-image encryption scheme combining DWT-based compressive sensing with discrete fractional random transform," *Opt. Commun.,* pp. 1–10, 2015, doi : 10.1016/j.optcom.2015.05.043.

[18] **R. Kumar and B. Bhaduri**, "Double image encryption in Fresnel domain using wavelet transform , gyrator transform and spiral phase masks," Fifth International Conference on optical and photonics Engineering, vol. 10449, pp. 4–9, 2017, doi : 10.1117/12.2269897.

[19] **H. Nematzadeh, R. Enayatifar, and H. Motameni**, "Medical image encryption using a hybrid model of modified genetic algorithm and coupled map lattices," *Opt. Lasers Eng.,* vol. 110, no. May, pp. 24–32, 2018, doi : 10.1016/j.optlaseng.2018.05.009.

[20] **S. Aashiq Banu and R. Amirtharajan**, "Tri-level scrambling and enhanced diffusion for DICOM image cipher- DNA and chaotic fused approach,"*Multimed. Tools Appl.,* vol. 79, no. 39–40, pp. 28807–28824, 2020, doi : 10.1007/s11042-020-09501-5.

[21] **X. Zhang and X. Wang**, "Multiple-image encryption algorithm based on DNA encoding and chaotic system," *Multimed. Tools Appl.,* vol. 78, pp. 7841-7869, 2018.

[22] **R. Enayatifar, A. Hanan, and I. F. Isnin**, "Chaos-based image encryption using a hybrid genetic algorithm and a DNA sequence," *Opt. Lasers Eng.,* vol. 56, pp. 83–93, 2014, doi : 10.1016/j.optlaseng.2013.12.003.

[23] **A. Belazi, M. Talha, S. Kharbech, W. E. I. Xiang, and S. Member**, "Novel Medical Image Encryption Scheme Based on Chaos and DNA Encoding," *IEEE Access,* vol. 7, pp. 36667–36681, 2019, doi : 10.1109/ACCESS.2019.2906292.

[24] **R. Hamza and K. Muhammad** "Hash Based Encryption for Keyframes of Diagnostic Hysteroscopy," vol. 5, 2018, doi : 10.1109/ACCESS.2017.2762405.

[25] **M. Boussif, N. Aloui, and A. Cherif**, "Smartphone application for medical images secured exchange based on encryption using the matrix product and the exclusive addition," *IET Image Process.,* vol. 11, no. 11, pp. 1020–1026, 2017, doi : 10.1049/iet-ipr.2017.0229.

[26] **K. Dharavathu and S. Anuradha**, *Efficient Transmission of an Encrypted Image Through a MIMO – OFDM System with Different Encryption Schemes.* Springer US, 2020.

[27] **M. A. Murillo-escobar, C. Cruz-hernández, and F. Abundiz-pérez**, "A RGB image encryption algorithm based on total plain image characteristics and chaos," *Signal Processing,* vol. 109, pp. 119–131, 2015, doi : 10.1016/j.sigpro.2014.10.033.

[28] **M. Hamdi, R. Rhouma, and S. Belghith**, "A Selective Compression-Encryption Of Images Based On SPIHT Coding and Chirikov Standard MAP," *Signal Processing,* 2016, doi : 10.1016/j.sigpro.2016.09.011.

[29] **S. Thakur, A. K. Singh, and S. P. Ghrera**, "Multi-layer security of medical data through watermarking and chaotic encryption for tele-health applications," Multimedia Tools and Appl., pp. 3457–3470, 2019.

[30] **X. Xie and Chin-Chen, Chang,** "Reversible Data Hiding in Encrypted Images Using Reformed JPEG Compression," *5th International Workshop on Biometrics and Forensics (IWBF),* pp. 0–4, 2017.

[31] **J. Chen, F. Han, W. Qian, and Y. Y. Z. Zhu**, "Cryptanalysis and improvement in an image encryption scheme using combination of the 1D chaotic map," *Nonlinear Dyn.,* vol. 93, no. 4, pp. 2399–2413, 2018, doi : 10.1007/s11071-018-4332-9.

[32] **Y. Chen, C. Tang, and R. Ye**, "Cryptanalysis and improvement of medical image encryption using high-speed scrambling and pixel adaptive diffusion," *Signal Processing,* p. 107286, 2019, doi : 10.1016/j.sigpro.2019.107286.

[33] **X. Wang and L. Liu**, "Cryptanalysis of a parallel sub-image encryption method with high-dimensional chaos," Nonlinear Dyn., pp. 795–800, 2013, doi : 10.1007/s11071-013-0832-9.

[34] **E. Y. Xie, C. Li, and S. Yu**, "On the cryptanalysis of Fridrich's chaotic image encryption scheme," *Signal Processing,* 2016, doi : 10.1016/j.sigpro.2016.10.002.

[35] **M. Li, Y. Guo, J. Huang, and Y. Li**, "Cryptanalysis of a chaotic image encryption scheme based on permutation-diffusion structure," *Signal Process. Image Commun.,* 2018, doi : 10.1016/j.image.2018.01.002.

[36] **H. Fan, M. Li, D. Liu, and K. An**, "Cryptanalysis of a plaintext-related chaotic RGB image encryption scheme using total plain image characteristics," Multimedia Tools and Appl., pp. 20103–20127, 2018.

[37] **M. Ahmad, E. Al Solami, X. Wang, M. N. Doja, and M. M. S. Beg**, "SS symmetry Cryptanalysis of an Image Encryption Algorithm Based on Combined Chaos for a BAN System , and Improved Scheme Using SHA-512 and Hyperchaos," Symmetry, pp. 1–18, 2018, doi : 10.3390/sym10070266.

[38] **R. Rhouma and S. Belghith**, "Cryptanalysis of a spatiotemporal chaotic image / video cryptosystem," Physics letters A, vol. 372, pp. 5790–5794, 2008, doi : 10.1016/j.chaos.2007.10.054.

[39] **B. Mirzakuchaki, SattaNorouzi and R**, "Breaking an Image Encryption Algorithm based on the New Substitution Stage with Chaotic Functions," *Opt. - Int. J. Light Electron Opt.,* 2016, doi : 10.1016/j.ijleo.2016.03.076.

[40] **Y. Liu**, "Cryptanalyzing a RGB image encryption algorithm based on DNA encoding and chaos map." *Opt. Laser Technol.,* vol. 60, pp. 111–115, 2014.

[41] **Y. Liu, H. Fan, and E. Y. Xie**, "Deciphering an image cipher based on mixed transformed Logistic maps," International Journal of bifurcation and chaos, vol. 25, no. 13, pp. 1–9, 2015, doi : 10.1142/S0218127415501886.

[42] **D. Arroyo, J. Diaz, and F. B. Rodriguez**, "Cryptanalysis of a one round chaos-based Substitution Permutation Network," *Signal Processing,* vol. 93, no. 5, pp. 1358–1364, 2013, doi : 10.1016/j.sigpro.2012.11.019..

[43] **Z. Liu, E. Blasch, G. Bhatnagar, V. John, W. Wu, and R. S. Blum**, "Fusing synergistic information from multi-sensor images : An overview from implementation to performance assessment," *Inf. Fusion,* vol. 42, no. October 2017, pp. 127–145, 2018, doi : 10.1016/j.inffus.2017.10.010.

[44] **M. Pradeep, "Implementation of image fusion algorithm using MATLAB (LAPLACIAN PYRAMID)**, *" Proc. - 2013 IEEE Int. Multi Conf. Autom. Comput. Control. Commun. Compress. Sensing, iMac4s 2013,* pp. 165–168, 2013, doi : 10.1109/iMac4s.2013.6526401.

[45] **S. Li, X. Kang, L. Fang, J. Hu, and H. Yin**, "Pixel-level image fusion : A survey of the state of the art," *Inf. Fusion,* vol. 33, pp. 100–112, 2017, doi : 10.1016/j.inffus.2016.05.004.

[46] **J. Suthakar**, "Study of Image Fusion-Techniques, Method and Applications," *Int. J. Comput. Sci. Mob. Comput.,* vol. 3, no. 11, pp. 469–476, 2014, [Online]. Available : www.ijcsmc.com.

[47] **Y. Yang, D. S. Park, S. Huang, and N. Rao**, "Medical image fusion via an effective wavelet-based approach," *EURASIP J. Adv. Signal Process.,* vol. 2010, 2010, doi : 10.1155/2010/579341.

[48] **X. Liu, W. Mei, and H. Du**, "Simultaneous image compression, fusion and encryption algorithm based on compressive sensing and chaos," *Opt. Commun.,* vol. 366, pp. 22–32, 2016, doi : 10.1016/j.optcom.2015.12.024.

[49] **Y. Liu, X. Tong, and J. Ma**, "Image encryption algorithm based on hyper-chaotic system and dynamic S-box," *Multimed. Tools Appl.,* vol. 75, no. 13, pp. 7739–7759, 2016, doi : 10.1007/s11042-015-2691-5.

[50] **Y. Qin, Q. Gong, Z. Wang, and H. Wang**, "Optical multiple-image encryption in diffractive-imaging-based scheme using spectral fusion and nonlinear operation," *Opt. Express,* vol. 24, no. 23, p. 26877, 2016, doi : 10.1364/oe.24.026877.

[51] **I. Mehra and N. K. Nishchal**, "Wavelet-based image fusion for securing multiple images through asymmetric keys," *Opt. Commun.,* vol. 335, pp. 153–160, 2015, doi :10.1016/j.optcom.2014.09.040.

[52] **S. Dongfeng et al.**, "Simultaneous fusion, imaging and encryption of multiple objects using a single-pixel detector," *Sci. Rep.,* vol. 7, no. 1, p. 13172, 2017, doi : 10.1038/s41598-017-12664-1.

[53] **A. Alfalou and A. Mansour**, "Double random phase encryption scheme to multiplex and simultaneous encode multiple images," *Optical Socity of America,* vol. 48, no. 31, 2009.

[54] **M. Alfalou, AymJridi and An**, "Real-time and encryption efficiency improvements of simultaneous fusion , compression and encryption method based on chaotic generators," *Opt. Lasers Eng.,* vol. 102, no. October 2017, pp. 59–69, 2018, doi : 10.1016/j.optlaseng.2017.10.007.

[55] "Simultaneous fusion, compression, and encryption of multiple images," *Opt. Express,* vol. 19, no. 24, p. 24023, 2011, doi : 10.1364/oe.19.024023.

[56] **X. Zhang and X. Wang**, "Multiple-image encryption algorithm based on mixed image element and permutation," *Opt. Lasers Eng.,* vol. 92, no. November 2016, pp. 6–16, 2017, doi : 10.1016/j.optlaseng.2016.12.005.

[57] **X. Zhang and X. Wang**, "Multiple-image encryption algorithm based on mixed image element and chaos," *Comput. Electr. Eng.,* vol. 62, pp. 401–413, 2017, doi : 10.1016/j.compeleceng.2016.12.025.

[58] **G.-L. Zhu**, "Mixed image element encryption algorithm based on an elliptic curve cryptosystem," *J. Electron. Imaging,* vol. 17, no. 2, p. 023007, 2008, doi : 10.1117/1.2931495.

[59] **A. M. Abdalla and A. A. Tamimi**, "Algorithm for Image Mixing and Encryption," *The International Journal of Multimedia & Its Applications (IJMA),* vol. 5, no. 2, pp. 15–21, 2013.

[60] **J. Chen, Z. Zhu, C. Fu, H. Yu, and Y. Zhang**, "Reusing the permutation matrix dynamically for efficient image cryptographic algorithm," *Signal Processing,* vol. 111, pp. 294–307, 2015, doi : 10.1016/j.sigpro.2015.01.003.

[61] **O. Mirzaei, M. Yaghoobi, and H. Irani**, "A new image encryption method : parallel sub-image encryption with hyper chaos," *Nonlinear Dynamics,* vol. 67, pp. 557–566, 2012, doi : 10.1007/s11071-011-0006-6

[62] **A. Alfalou, C. Brosseau, N. Abdallah, and M. Jridi**, "Assessing the performance of a method of simultaneous compression and encryption of multiple images and its resistance against various attacks," Optics Express, vol. 21, no. 7, pp. 10253–10265, 2013.

[63] **A. Alfalou, I. S. Member, and B. Cedex**, "New Image Encryption Method Based on ICA," *MVA2007 IAPR Conference on Machine Vision Applications,* pp. 223–226, 2007.

[64] **Douglas Stinson, D.** "Cryptography Theory and practical," Traduction of the Serge Vaudenay, Gildas Avoine and Pascal Junod (2 nd Edition). Paris Vuibert Informatics, 2003.

[65] **R. L. Rivest, A. Shamir, and L. Adleman**, "A Method for Obtaining Digital Signatures and Public-Key Cryptosystems Public-Key Cryptosystems." vol. 21, no. 2, 1978.

[66] **R. Dumont.**, Cryptographie et Sécurité informatique, Université de liége faculté des sciences appliquées 2007, https ://www.ulg.ac.be.

[67] **S. Ramakrishnan**, "Cryptographic and Information Security Approaches for Images and Videos," CRC Press is an imprint of the Taylor & Francis Group, an informa business, 2019.

[68] **M. BOUKHATEM Mohammed Belkaid**, "Application des techniques de cryptage pour la transmission sécurisée d'images MSG," Mémoire de magister en électronique option : Télédétection, Université Mouloud Mammeri, Tizi-Ouzou faculté de génie électrique et de l'informatique Département d'électronique, 2015.

[69] **S. M. Seyedzadeh, B. Norouzi, et S. Mirzakuchaki**, "RGB color image coding based on the scrambled Choquet integral," *system log and software,* vol. 97, pp.128 - 139, 2014.

[70] **S. Ramakrishnan**, *Cryptographic and Information Security.* 2018.

[71] S, "Communication theory of secrecy systems. 1945.," *MD. Comput.,* vol. 15, no. 1, pp. 57–64, 1998.

[72] **Y. Abanda, A. Tiedeu, and G. Kom**, "Image Encryption with Fusion of Two Maps," *Security and Communication Networks,* vol. 2021, 2021.

[73] **J.-G. Dumas, J.-L. Roch, É. Tannier, and S. Varrette**, *Théorie des codes.* Dunod, Paris, 2007.

[74] **Bruce schneier,** "applied cryptography," CRC press, united states of America, 1996, p. 780.

[75] **R. stinson,** "cryptography : theory and practice, (discrete mathematics and its applications)," chapman & hall/ crc press, new york, november 2005.

[76] **C. Pohl and J. L. Van Genderen**, *Review article Multisensor image fusion in remote sensing : Concepts, methods and applications,* vol. 19, no. 5. 1998.

[77] **M. G. Djaouher**, "Comparatif de méthodes avancées pour la fusion d'images satellites," Thèse de doctorat PhD, Rayonnement et matière, Université d'ORAN des sciences et de la technologie USTD-MB, faculté de Physique, département de Genie physique, 2015.

[78] **P. S. Chavez, S. C. Sides, and J. A. Anderson**, "Comparison of three different methods to merge multiresolution and multispectral data : Landsat TM and SPOT panchromatic," *Photogramm. Eng. Remote Sens.,* vol. 57, no. 3, pp. 295–303, 1991, doi : 10.1306/44b4c288-170a-11d7-8645000102c1865d.

[79] **S. Jaryal**, "Comparative Analysis of Various İmage Encryption Techniques," *Int. J. Comput. Intell. Res.,* vol. 13, no. 2, pp. 273–284, 2017.

[80] **V. P. S. Naidu and J. R. Raol**, "Pixel-level image fusion using wavelets and principal component analysis," *Def. Sci. J.,* vol. 58, no. 3, pp. 338–352, 2008, doi : 10.14429/dsj.58.1653.

[81] **S. Li, J. T. Kwok, and Y. Wang**, "Multifocus image fusion using artificial neural networks," Pattern recognition letters, vol. 23, pp. 985–997, 2002.

[82] **Z. Liu and S. Liu**, "Double image encryption based on iterative fractional Fourier transform," *Opt. Commun.,* vol. 275, no. 2, pp. 324–329, 2007, doi : 10.1016/j.optcom.2007.03.039.

[83] **A. Mansour and M. Kawamoto**, "ICA papers classified according to their applications and performances," *IEICE Trans. Fundam. Electron. Commun. Comput. Sci.,* vol. E86-A, no. 3, pp. 620–633, 2003.

[84] **Mohammed ALDOSSARI**, "Nouvelle méthode optique de compression et de cryptage simultanés des images (fixes/vidéo) pour les systèmes télécommunication," Thèse de Doctorat, École Doctorale SICMA, Université de Bretagne Occidentale, 2014.

[85] **W. M. H. Company**, *Modern Cryptography : Theory and Practice.* 2003.

[86] **A. A. Karawia**, "Encryption Algorithm of Multiple-Image Using Mixed Image Elements and Two Dimensional Chaotic Economic Map," *Entropy,* vol. 20, pp. 801, 2018, doi : 10.3390/e20100801.

[87] **A. Alfalou, C. Brosseau, N. Abdallah, and M. Jridi**, "Simultaneous fusion , compression , and encryption of multiple images," vol. 19, no. 24, pp. 24023–24029, 2011.

[88] **X. Zhang and X. Wang**, "Digital Image Encryption Algorithm Based on Elliptic Curve Public Cryptosystem," *IEEE Access,* vol. 6, pp. 70025–70034, 2018, doi : 10.1109/ACCESS.2018.2879844.

[89] **Y. Zhou, L. Bao, and C. L. P. Chen**, "Signal Pro- cessing A New 1D Chaotic System for Image Encryption," no. May 2019, 2014, doi : 10.1109/ICSSE.2012.6257151.

[90] **Y. Pascal, K. Nkandeu, and A. Tiedeu,**, "An image encryption algorithm based on substitution technique and chaos mixing," *Multimed. Tools Appl.,* vol. 78(8), pp. 10013-10034, 2018, doi : 10.1007/s11042-018-6612-2.

[91] **L. Xu, Z. Li, J. Li, and W. Hua**, "A novel bit-level image encryption algorithm based on chaotic maps," Signal Process., vol. 78, pp. 17–25, 2016, doi : 10.1016/j.optlaseng.2015.09.007.

[92] **K. K. Butt, G. Li, and S. Khan**, "Fast and Efficient Image Encryption Algorithm Based," Entropy, pp. 1–28, 2020, doi : 10.3390/e22010112.

[93] **A. Belazi and A. A. A. El-latif**, "Author ' s Accepted Manuscript," *Signal Processing,* 2016, doi : 10.1016/j.sigpro.2016.03.021.

[94] **A. Kanso and M. Ghebleh**, "Commun Nonlinear Sci Numer Simulat An efficient and robust image encryption scheme for medical applications," *Commun. NONLINEAR Sci. Numer. Simul.,* vol. 24, no. 1–3, pp. 98–116, 2015, doi : 10.1016/j.cnsns.2014.12.005.

[95] "A novel image encryption scheme based on improved random number generator and its implementation," pp. 1781–1805, 2019, doi : 10.1007/s11071-018-4659-2.

[96] **H. Hamiche, S. Guermah, R. Saddaoui, K. Hannoun, M. Laghrouche, and S. Djennoune**, "Analysis and implementation of a novel robust transmission scheme for private digital communications using Arduino Uno board," *Nonlinear Dyn.,* pp. 1921-1932, 2015, doi : 10.1007/s11071-015-2116-z.

[97] **M. K. Mandal and A. K. Das**, "Chaos-Based Colour Image Encryption Using Microcontroller ATMEGA 32," *Nanoelectronics, Circuits and Communication Systems, Lecture Notes in Electrical Engineering,* pp. 281-287, 2018.

[98] **A. K. D. S. H. M. K. Mandal**, "RGB image encryption using microcontroller ATMEGA 32," *Microsyst. Technol.,* vol. 5, 2018, doi : 10.1007/s00542-018-3980-5.

[99] **F. Sahib, H. Maryam, and A. Saffo**, "FPGA Hardware Co - Simulation of Image Encryption Using Stream Cipher Based on Chaotic Maps," *Sens. Imaging,* 2020, doi : 10.1007/s11220-020-00301-7.

[100] **R. Chiu, M. Mora-gonzalez, and D. Lopez-mancilla**, "Implementation of a Chaotic Oscillator into a Simple Microcontroller," *IERI Procedia,* vol. 4, pp. 247–252, 2013, doi : 10.1016/j.ieri.2013.11.035.

[101] **C. Fu et al.**, "An ef fi cient and secure medical image protection scheme based on chaotic maps," *Comput. Biol. Med.,* vol. 43, no. 8, pp. 1000–1010, 2013, doi : 10.1016/j.compbiomed.2013.05.005.

[102] **X. Wang and Q. Wang**, "A novel image encryption algorithm based on dynamic S-boxes constructed by chaos," *Nonlinear Dynamics,* vol. 75, pp. 567–576, 2014.

[103] **Z. Hua, S. Yi, and Y. Zhou**, "Medical image encryption using high-speed scrambling and pixel adaptive diffusion," Signal Process., vol. 144, pp. 134–144, 2018, doi : 10.1016/j.sigpro.2017.10.004.

[104] **K. Shankar, M. Elhoseny, E. D. Chelvi, S. K. Lakshmanaprabu, and W. Wu**, "An Efficient Optimal Key Based Chaos Function for Medical Image Security," *IEEE Access,* vol. 6, pp. 77145–77154, 2018, doi : 10.1109/ACCESS.2018.2874026.

[105] **J. Liu and Y. Ma**, "A new simple chaotic system and its application in medical image encryption," *Multimed. Tools and Appl.,* vol. 77, pp. 22787–22808, 2018.

[106] **S. Kumar, B. Panna, and R. Kumar**, "Medical image encryption using fractional discrete cosine transform with chaotic function," *Medical and Biological Engineering and Computing,* vol. 57, pp. 2517–2533, 2019.

[107] **M. Benssalah and Y. Rhaskali**, "Medical Images Encryption Based on Elliptic Curve Cryptography and Chaos Theory," *2018 Int. Conf. Smart Commun. Netw. Technol.,* pp. 222–226, 2018.

[108] **A. Banu S and R. Amirtharajan**, "A robust medical image encryption in dual domain : chaos-DNA-IWT combined approach," *Med. Biol. Eng. Comput.,* vol. 58, no. 7, pp. 1445–1458, 2020, doi : 10.1007/s11517-020-02178-w.

[109] **G. Ren, J. Han, H. Zhu, J. Fu, and M. Shan**, "High Security Multiple-image Encryption using Discrete Cosine Transform and Discrete Multiple-Parameter Fractional Fourier Transform," vol. 11, no. 5, 2016, doi : 10.12720/jcm.11.5.491-497.